



User Guide

AWS Elemental MediaTailor



AWS Elemental MediaTailor: User Guide

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Die Handelsmarken und Handelsaufmachung von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, durch die Kunden irregeführt werden könnten oder Amazon in schlechtem Licht dargestellt oder diskreditiert werden könnte. Alle anderen Handelsmarken, die nicht Eigentum von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise zu Amazon gehören oder nicht, mit Amazon verbunden sind oder von Amazon gesponsert werden.

Table of Contents

Was ist AWS Elemental MediaTailor?	1
MediaTailor Konzepte	1
Konzepte zur Anzeigeneinfügung	1
Konzepte zur Kanalmontage	3
Wie funktioniert das Einfügen von MediaTailor Anzeigen	4
Zugehörige Services	5
Zugreifen MediaTailor	5
Preisgestaltung	6
Regionen	6
Einrichtung	7
Melden Sie sich an für ein AWS-Konto	7
Erstellen eines Benutzers mit Administratorzugriff	8
Erste Schritte mit MediaTailor	10
Erste Schritte mit dem Einfügen MediaTailor von Anzeigen	10
Voraussetzungen	11
Schritt 1: Zugriff MediaTailor	11
Schritt 2: Bereite einen Stream vor	11
Schritt 3: Konfigurieren Sie die ADS-Anforderungs-URL und die Abfrageparameter	13
Schritt 4: Erstellen einer Konfiguration	14
Schritt 5: Testen Sie die Konfiguration	16
Schritt 6: Senden Sie die Wiedergabeanforderung an MediaTailor	18
(Optional) Schritt 7: Aktivität überwachen	19
Schritt 8: Bereinigen	20
Erste Schritte mit der MediaTailor Kanalmontage	21
Voraussetzungen	22
Schritt 1: Erstellen Sie einen Quellspeicherort	22
Schritt 2: Fügen Sie VOD-Quellen zu Ihrem Quellstandort hinzu	23
Schritt 3: Erstellen Sie einen Kanal	24
Schritt 4: Füge Programme zum Zeitplan deines Kanals hinzu	26
Schritt 5 (optional): Verwenden Sie diese Option MediaTailor , um personalisierte Anzeigen in Ihren Stream einzufügen	28
Schritt 6: Starte deinen Kanal	29
Schritt 7: Teste deinen Kanal	29
Schritt 8: Bereinigen	29

Anzeigen einfügen	30
Unterstützte Audio- und Videocodecs	31
Das Verhalten bei der Insertion von Anzeigen verstehen	31
Verhalten beim Zusammenfügen von Anzeigen bei VOD	31
Verhalten beim Zusammenfügen von Live-Anzeigen	34
Grundlegendes zur servergesteuerten Anzeigeneinfügung	36
In der Wiedergabekonfiguration aktivieren	36
Erstellen Sie eine servergeführte Sitzung	36
Anforderungen für Adserver-Integrationen	38
RIESIGE Anforderungen	39
VPAID-Anforderungen	41
Mit Wiedergabekonfigurationen arbeiten	42
Konfiguration erstellen	43
Konfiguration anzeigen	51
Eine Konfiguration bearbeiten	52
Eine Konfiguration löschen	52
Integration einer Inhaltsquelle	53
Anforderungen an die Eingabequelle	53
Integration einer HLS-Quelle	53
Integration einer MPEG-DASH-Quelle	63
Sicherung der ursprünglichen Interaktionen mit SigV4	124
Integration mit Google Ad Manager	128
Serverseitige Integration	129
Kundenseitige Integration	130
Verwenden eines CDN	131
Integration eines CDN	132
Wie MediaTailor geht man mit Base URLs for DASH um	136
Bewährte Methoden für CDN	136
Anpassen des Verhaltens von Werbeunterbrechungen mit Anzeigenunterdrückung	137
Die Unterdrückung von Werbeunterbrechungen konfigurieren	137
Stoßstangen einsetzen	152
Stoßstangen konfigurieren	153
Pre-Roll-Anzeigen einfügen	154
Schiefer wird eingefügt	155
Das Slate konfigurieren	156
Slate-Konfiguration und VPAID	156

Anzeigen vorab abrufen	156
So funktioniert der Vorabruf	158
Prefetch-Zeitpläne erstellen	162
Löschen von Prefetch-Zeitplänen	168
Vorkonditionierte Anzeigen	169
Anforderungen an vorkonditionierte Anzeigen	169
Arbeitsablauf für vorkonditionierte Anzeigen	173
Verwendung dynamischer Anzeigenvariablen	174
Übergabe von Parametern an das ADS	174
Verwendung von Domänenvariablen	179
Verwenden von Sitzungsvariablen	182
Verwendung von Spielervariablen	195
Übergabe von Abfrageparametern an das Manifest	198
Implizite HLS-Sitzungsinitialisierung	199
Implizite DASH-Sitzungsinitialisierung	200
Explizite HLS- und DASH-Sitzungsinitialisierung	201
Daten zur Berichterstattung und Nachverfolgung	202
Serverseitiges Tracking	203
Kundenseitiges Tracking	204
Overlay-Anzeigen	301
Voraussetzungen für die Verwendung von Overlay-Anzeigen	302
Erste Schritte	302
Protokollierung und Metriken	316
Abrechnung für Overlay-Anzeigen in MediaTailor	318
ID-Dekoration hinzufügen	318
Status der Sitzung	319
Manifeste und Einfügen von Anzeigenmetadaten	321
Interaktionen mit dem Ad Decision Server (ADS)	345
Client-seitige Tracking-API	346
Linear zusammengestellte Streams erstellen	348
Mit Quellstandorten arbeiten	348
Einen Quellspeicherort erstellen	349
Konfiguration der Authentifizierung für Ihren Quellstandort	351
Mit VOD-Quellen arbeiten	361
Arbeiten mit Live-Quellen	365
Verwenden von Paketkonfigurationen	370

Zwischenspeichern von Manifesten	370
Mit Kanälen arbeiten	371
Erstellen Sie einen Kanal	371
Verwende Quellgruppen mit den Ausgaben deines Kanals	374
Löscht einen Kanal	374
Ein Programm hinzufügen	375
Ein Programm erstellen	375
Definition von Zielgruppenkohorten und alternativen Inhalten	383
Generierung zielgruppenspezifischer Manifeste	386
Fügen Sie Anzeigen und Werbeunterbrechungen ein	387
Anzeigeneinfügung einrichten	387
SCTE-35-Nachrichten für Werbeunterbrechungen	388
Zeitversetzte Anzeige aktivieren	395
Zeitversetzende Parameter für Manifestanfragen	396
Verwenden Sie zeitversetzte Anzeige mit CDNs	399
Behebung von Wiedergabefehlern	401
Client-Fehler	402
Serverfehler	404
Beispiele	406
Sicherheit	408
Datenschutz	409
Datenverschlüsselung	410
Identitäts- und Zugriffsverwaltung	410
Zielgruppe	411
Authentifizierung mit Identitäten	411
Verwalten des Zugriffs mit Richtlinien	415
Wie AWS Elemental MediaTailor funktioniert mit IAM	418
Beispiele für identitätsbasierte Richtlinien	425
Beispiele für eine ressourcenbasierte Richtlinie	428
AWS verwaltete Richtlinien	430
Verwenden von serviceverknüpften Rollen	431
Fehlerbehebung für -Identität und -Zugriff	435
Compliance-Validierung	437
Ausfallsicherheit	438
Sicherheit der Infrastruktur	438
Serviceübergreifende Confused-Deputy-Prävention	439

Protokollierung und Überwachung	441
CloudWatch Alarme	441
CloudTrail Logs	441
AWS Trusted Advisor	441
Überwachung und Tagging	443
Anzeigen von -Protokollen	444
ADS-Protokolle	444
Manifest-Protokolle	495
Protokolle transkodieren	500
Verwendung von verkauften Protokollen	504
Protokolle in CloudWatch Logs schreiben	509
Steuerung des Volumens der Sitzungsprotokolle zur Anzeigenschaltung	519
Filterung von Protokollen und Ereignissen	522
Debug-Logs generieren	526
Überwachung mit CloudWatch Metriken	531
AWS Elemental MediaTailor CloudWatch Metriken	532
AWS Elemental MediaTailor CloudWatch Abmessungen	542
Verwendung von Metriken zur Diagnose veralteter Manifeste	542
API-Aufrufe aufzeichnen	545
AWS Elemental MediaTailor Informationen in CloudTrail	546
Grundlagen zu AWS Elemental MediaTailor -Protokolldateieinträgen	547
Empfangen von Channel Assembly-Benachrichtigungen	549
Warnmeldungen anzeigen	555
Umgang mit Warnmeldungen	556
Taggen von -Ressourcen	556
Unterstützte Ressourcen	557
Tag-Einschränkungen	557
Verwalten von Tags	558
Workflow-Monitor	558
Komponenten des Workflow-Monitors	560
Unterstützte Services	560
Workflow-Monitor konfigurieren	561
Verwenden Sie den Workflow-Monitor	581
Kontingente	584
Kontingente für das Einfügen von Anzeigen	584
Kontingente für die Kanalmontage	589

MediaTailor Ressourcen 595

Dokumentverlauf 597

..... dcxvi

Was ist AWS Elemental MediaTailor?

AWS Elemental MediaTailor ist ein skalierbarer Anzeigeneinfügungs- und Channel-Assembly-Service, der in der AWS Cloud ausgeführt wird. Mit MediaTailor können Sie gezielte Anzeigeninhalte für Zuschauer bereitstellen und lineare Streams erstellen, während die Broadcast-Qualität in over-the-top (OTT) -Videoanwendungen gewahrt wird. MediaTailor Die Anzeigeneinfügung unterstützt Apple HTTP Live Streaming (HLS) und MPEG Dynamic Adaptive Streaming over HTTP (DASH) für Video-on-Demand (VOD) und Live-Workflows.

AWS Elemental MediaTailor Das Einfügen von Werbeanzeigen bietet wichtige Fortschritte gegenüber herkömmlichen Ad-Tracking-Systemen: Anzeigen lassen sich besser monetarisieren, haben eine konsistentere Videoqualität und Auflösung und sind in plattformübergreifenden Umgebungen einfacher zu verwalten. MediaTailor vereinfacht Ihren Werbe-Workflow, indem es allen Geräten mit IP-Verbindung ermöglicht wird, Anzeigen auf die gleiche Weise zu rendern wie andere Inhalte. Der Service bietet auch erweiterte Nachverfolgung von Werbe-Aufrufen, womit sich die Monetisierung von Inhalten weiter erhöht.

AWS Elemental MediaTailor Channel Assembly ist ein reiner Dienst, mit dem Sie lineare Streaming-Kanäle mithilfe Ihrer vorhandenen Video-on-Demand-Inhalte (VOD) erstellen können. MediaTailor berührt niemals Ihre Inhaltssegmente, die direkt von Ihrem Ursprungsserver bereitgestellt werden. MediaTailor ruft stattdessen die Manifeste von Ihrem Ursprung ab und verwendet sie, um ein Live-Slide-Manifestfenster zusammenzustellen, das auf die zugrunde liegenden Inhaltssegmente verweist.

MediaTailor Die Kanalzusammenstellung macht es einfach, deinen Kanal zu monetarisieren, indem du Werbeunterbrechungen in deinen Stream einfügst, ohne dass du ihn mit SCTE-35-Markern versehen musst. Du kannst Channel Assembly mit MediaTailor Anzeigeneinfügung oder einen anderen serverseitigen Dienst zur Anzeigeneinfügung verwenden.

MediaTailor Konzepte

Im Folgenden finden Sie eine Übersicht über die Konzepte, die in der AWS Elemental MediaTailor Bedienungsanleitung verwendet werden.

Konzepte zur Anzeigeneinfügung

Im Folgenden finden Sie eine Übersicht über die Konzepte im Zusammenhang mit der Anzeigeneinfügung.

Und verfügbar

Eine bestimmte Werbezeiteinheit innerhalb einer Werbepause, die an Werbetreibende verkauft werden kann. Eine Werbeunterbrechung kann mehrere verfügbare Werbeanzeigen enthalten. Wenn MediaTailor eine VAST-Antwort von einem Ad Decision Server erhält, füllt er diese Verfügbarkeiten mit personalisierten Anzeigen.

Werbepause

Der Zeitraum während der Programmierung, in dem Werbespots gezeigt werden. Werbeunterbrechungen können vor dem Inhalt (Pre-Roll), während des Inhalts (Mid-Roll) oder nach dem Inhalt (Post-Roll) auftreten. MediaTailor identifiziert Werbeunterbrechungen in Manifesten anhand von Markierungen wie SCTE-35.

Anzeigen-Entscheidungsserver (ADS)

Ein Server, der Werbespot-Spezifikationen basierend auf Kriterien wie z. B. aktuelle Werbekampagnen und Betrachtervorzügen bereitstellt.

Konfiguration

Ein Objekt MediaTailor, mit dem Sie interagieren. Die Konfiguration enthält Standortinformationen über den Ursprungsserver und den Anzeigen-Entscheidungsserver (ADS). Die Konfiguration enthält auch Endpunkte, die Eingangs- und Ausgangspunkte bereitstellen. MediaTailor

Dynamisches Transkodierung

Ein Prozess, der Qualität und Format der Werbung an die primären Videoinhalte anpasst, wenn Inhalte angefordert werden. Dynamische Transcodierung reduziert die Speicheranforderungen und sorgt für einen nahtlosen Übergang der Wiedergabe zwischen Werbung und Videoinhalten.

Manifeste Manipulation

Bei diesem Prozess werden Manifeste vom Ursprungs-Server so umgeschrieben, dass sie auf die entsprechenden Werbe- und Inhalts-Fragmente verweisen. Anzeigen werden durch die VAST-Antwort des Ad Decision Server (ADS) bestimmt. Fügt im Verlauf der Wiedergabe MediaTailor die Anzeige ein oder ersetzt sie in den Inhaltsstream.

VAST und VMAP

Video Ad Serving Template (VAST) und Video Multiple Ad Playlist (VMAP) sind XML-Antworten, die der Ad Decision Server auf Anzeigenanfragen sendet, von denen aus MediaTailor Die Antworten bestimmen, welche Anzeigen in das MediaTailor Manifest eingefügt werden. VMAP beinhaltet auch den Zeitpunkt für Werbeunterbrechungen und die darin enthaltene Verfügbarkeit

von Werbeanzeigen. Weitere Informationen zur Logik hinter der MediaTailor Anzeigeneinfügung finden Sie unter [Das Verhalten bei der AWS Elemental MediaTailor Anzeigeneinblendung verstehen](#). Weitere Informationen über die MediaTailor Funktionsweise von mit VAST finden Sie unter [the section called “Anforderungen für Adserver-Integrationen”](#).

Konzepte zur Kanalmontage

Im Folgenden finden Sie eine Übersicht über die Konzepte im Zusammenhang mit der Kanalmontage.

Kanäle

Ein Kanal fügt Ihre Quellmanifeste zu einem linearen Stream zusammen. Jeder Kanal hat einen oder mehrere Ausgänge, die die Wiedergabe enthalten, auf die Spieler URLs zugreifen. Die Kanalausgänge entsprechen den Paketkonfigurationseinstellungen, die Sie für Ihre VOD-Quellen erstellen. Ein Kanal enthält einen Zeitplan, der festlegt, wann VOD-Quellen im Stream des Kanals wiedergegeben werden.

Verpackungskonfiguration

Eine Packager-Konfiguration ist eine Darstellung Ihrer VOD-Quelle, die bestimmte Merkmale des Paketformats enthält. Sie verknüpfen Ihre Paketkonfigurationen mit Kanalausgängen, um Wiedergabestreams für die Paketformate Ihrer VOD-Quelle zu erstellen, z. B. HTTP Live Streaming (HLS).

Zeitplan

Jeder Kanal besteht aus Programmen, die im Zeitplan des Kanals angeordnet sind. Der Zeitplan bestimmt, zu welcher Uhrzeit die Programme im linearen Stream des Kanals abgespielt werden.

Quellstandorte

Ein Quellstandort stellt den Ursprungsserver dar, auf dem Ihre Ressourcen gespeichert sind. Es kann sich um Amazon S3, einen HTTP-Server, ein Content Delivery Network (CDN) oder eine Verpackungsinfrastruktur wie MediaPackage handeln.

VOD-Quellen

Eine VOD-Quelle steht für einen einzelnen Inhalt, z. B. einen Film oder eine Folge einer Fernsehsendung. Sie verknüpfen VOD-Quellen mit Programmen, um sie dem linearen Stream Ihres Kanals hinzuzufügen.

Zielgruppe

Eine Zielgruppe definiert eine Zuschauerkohorte, die optional alternative Inhalte haben kann. Sie können Zielgruppen auf linearen Standardkanälen definieren.

Wie funktioniert das Einfügen von MediaTailor Anzeigen

AWS Elemental MediaTailor interagiert zwischen Ihrem Content Delivery Network (CDN), Ihrem Ursprungsserver und Ihrem Ad Decision Server (ADS), um personalisierte Anzeigen in Werbeunterbrechungen innerhalb von Live- und Video-on-Demand-Inhalten einzubinden.

Im Folgenden finden Sie eine Übersicht darüber, wie die MediaTailor Anzeigeneinfügung funktioniert:

1. Ein Player oder CDN wie Amazon CloudFront sendet eine Anfrage MediaTailor für HLS- oder DASH-Inhalte an. Die Anfrage enthält Parameter des Players mit Informationen über den Zuschauer, die für die Personalisierung von Anzeigen verwendet werden.
2. MediaTailor sendet eine Anfrage an das ADS, die die Zuschauerinformationen enthält. Der ADS wählt Werbespots basierend auf den Informationen zum Betrachter und aktuellen Werbekampagnen aus. Sie gibt die URLs an die Werbekunden in einer VAST- oder VMAP-Antwort an zurück. MediaTailor

Wenn Sie die Anzeigen vorab konditioniert haben, URLs verweisen sie auf die vortranskodierte Anzeigen. Informationen zum Anzeigen-Stitching mit vortranskodierte Anzeigen finden Sie unter [Vorkonditionierte Anzeigen](#)

3. MediaTailor manipuliert das Manifest so, dass es die Anzeige enthält, die vom ADS URLs zurückgegeben wurde, und zwar transcodiert, sodass sie den Kodierungsmerkmalen des ursprünglichen Inhalts entspricht. Wenn Sie vorkonditionierte Anzeigen verwenden, liegt es in Ihrer Verantwortung, sicherzustellen, dass die Anzeige dem Vorlagenmanifest entspricht.

Wenn eine Anzeige noch nicht so transkodiert wurde, dass sie dem Inhalt entspricht, MediaTailor wird das Einfügen übersprungen und die Anzeige wird MediaConvert zur Vorbereitung der Anzeige verwendet, sodass sie für die nächste Anfrage bereit ist.

4. MediaTailor sendet das vollständig personalisierte Manifest an das anfragende CDN oder den Player zurück.

Das ADS verfolgt die aufgerufenen Anzeigen anhand von Meilensteinen wie dem Beginn der Anzeige, der Mitte der Anzeige und dem Ende der Anzeige. Im Verlauf der Wiedergabe MediaTailor sendet der Player oder der Player Werbe-Tracking-Beacons an die ADS-Anzeigen-Tracking-URL, um aufzuzeichnen, wie viel von einer Anzeige angesehen wurde. Bei der Sitzungsinitialisierung mit gibt der Player an MediaTailor, ob er diese Beacons für die Sitzung senden MediaTailor soll oder soll.

Weitere Informationen über die ersten Schritte mit der Anzeigeneinfügung finden Sie unter [Erste Schritte mit MediaTailor](#).

Zugehörige Services

- Amazon CloudFront ist ein globaler Netzwerkservice zur Bereitstellung von Inhalten (CDN), der sicher Daten und Videos für Zuschauer bereitstellt. Verwenden Sie CloudFront, um Inhalte mit der bestmöglichen Leistung bereitzustellen. Weitere Informationen zu CloudFront finden Sie auf der [CloudFrontAmazon-Website](#).
- AWS Elemental MediaPackage ist ein just-in-time Verpackungs- und Originalservice, der Live-Video-Assets für die Verteilung in einem Format anpasst, das mit dem Gerät, das die Anfrage stellt, kompatibel ist. Verwenden Sie AWS Elemental MediaPackage als Ursprungsserver, um Inhalte vorzubereiten und Werbemarker hinzuzufügen, bevor Sie Streams an MediaTailor senden. Weitere Informationen darüber, wie es mit MediaTailor Originalservern funktioniert, finden Sie unter [Wie funktioniert das Einfügen von MediaTailor Anzeigen](#).
- AWS Identity and Access Management (IAM) ist ein Webservice, der es Ihnen ermöglicht, den Zugriff auf AWS-Ressourcen für Ihre Benutzer sicher zu steuern. Kontrollieren Sie mit IAM, wer Ihre AWS-Ressourcen verwenden kann (Authentifizierung) und welche Ressourcen auf welche Weise verwendet werden können (Autorisierung). Weitere Informationen finden Sie unter [Einrichten AWS Elemental MediaTailor](#).

Zugreifen MediaTailor

Sie können MediaTailor über die Konsole des Dienstes darauf zugreifen.

Greifen Sie auf Ihre zu, AWS-Konto indem Sie Anmeldeinformationen angeben, mit denen bestätigt wird, dass Sie zur Verwendung der Services berechtigt sind.

Verwenden Sie den folgenden Link, um sich bei der MediaTailor Konsole anzumelden: **<https://console.aws.amazon.com/mediatailor/home>**.

Preisgestaltung für MediaTailor

Wie bei anderen AWS -Produkten gibt es keine vertraglichen Verpflichtungen oder Mindestlaufzeiten für die Verwendung MediaTailor. Ihnen werden Gebühren basierend auf Ihrer Nutzung des Service berechnet. Weitere Informationen finden Sie unter [MediaTailor Preise](#).

Regionen für MediaTailor

Zur Verkürzung der Datenlatenz in Ihren Anwendungen MediaTailor bietet es regionale Endpunkte zum Stellen Ihrer Anforderungen. Eine Liste der Regionen, in denen MediaTailor es verfügbar ist, finden Sie unter [Regionale Endpunkte](#).

Einrichten AWS Elemental MediaTailor

Dieser Abschnitt führt Sie durch die Schritte zum Konfigurieren von Benutzern für den Zugriff auf AWS Elemental MediaTailor. Hintergrundinformationen und zusätzliche Informationen zur Identitäts- und Zugriffsverwaltung für finden Sie MediaTailor unter [Identity and Access Management für AWS Elemental MediaTailor](#).

Führen Sie die folgenden Schritte aus AWS Elemental MediaTailor, um mit der Verwendung zu beginnen.

Themen

- [Melden Sie sich an für ein AWS-Konto](#)
- [Erstellen eines Benutzers mit Administratorzugriff](#)

Melden Sie sich an für ein AWS-Konto

Wenn Sie noch keine haben AWS-Konto, führen Sie die folgenden Schritte aus, um eine zu erstellen.

Um sich für eine anzumelden AWS-Konto

1. Öffnen Sie [https://portal.aws.amazon.com/billing/die Anmeldung](https://portal.aws.amazon.com/billing/die-Anmeldung).
2. Folgen Sie den Online-Anweisungen.

Ein Teil des Anmeldevorgangs umfasst den Empfang eines Telefonanrufs oder einer Textnachricht und die Eingabe eines Bestätigungscode auf der Telefontastatur.

Wenn Sie sich für eine anmelden AWS-Konto, wird eine Root-Benutzer des AWS-Kontos erstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Als bewährte Sicherheitsmethode weisen Sie einem Administratorbenutzer Administratorzugriff zu und verwenden Sie nur den Root-Benutzer, um [Aufgaben auszuführen, die Root-Benutzerzugriff erfordern](#).

AWS sendet Ihnen nach Abschluss des Anmeldevorgangs eine Bestätigungs-E-Mail. Du kannst jederzeit deine aktuellen Kontoaktivitäten einsehen und dein Konto verwalten, indem du zu <https://aws.amazon.com/> gehst und Mein Konto auswählst.

Erstellen eines Benutzers mit Administratorzugriff

Nachdem Sie sich für einen angemeldet haben AWS-Konto, sichern Sie Ihren Root-Benutzer des AWS-Kontos AWS IAM Identity Center, aktivieren und erstellen Sie einen Administratorbenutzer, sodass Sie den Root-Benutzer nicht für alltägliche Aufgaben verwenden.

Sichern Sie Ihre Root-Benutzer des AWS-Kontos

1. Melden Sie sich [AWS Management Console](#) als Kontoinhaber an, indem Sie Root-Benutzer auswählen und Ihre AWS-Konto E-Mail-Adresse eingeben. Geben Sie auf der nächsten Seite Ihr Passwort ein.

Hilfe bei der Anmeldung mit dem Root-Benutzer finden Sie unter [Anmelden als Root-Benutzer](#) im AWS-Anmeldung Benutzerhandbuch zu.

2. Aktivieren Sie die Multi-Faktor-Authentifizierung (MFA) für den Root-Benutzer.

Anweisungen finden Sie unter [Aktivieren eines virtuellen MFA-Geräts für Ihren AWS-Konto Root-Benutzer \(Konsole\)](#) im IAM-Benutzerhandbuch.

Erstellen eines Benutzers mit Administratorzugriff

1. Aktivieren Sie das IAM Identity Center.

Anweisungen finden Sie unter [Aktivieren AWS IAM Identity Center](#) im AWS IAM Identity Center Benutzerhandbuch.

2. Gewähren Sie einem Administratorbenutzer im IAM Identity Center Benutzerzugriff.

Ein Tutorial zur Verwendung von IAM-Identity-Center-Verzeichnis als Identitätsquelle finden Sie IAM-Identity-Center-Verzeichnis im Benutzerhandbuch unter [Benutzerzugriff mit der Standardeinstellung konfigurieren](#).AWS IAM Identity Center

Anmelden als Administratorbenutzer

- Um sich mit Ihrem IAM-Identity-Center-Benutzer anzumelden, verwenden Sie die Anmelde-URL, die an Ihre E-Mail-Adresse gesendet wurde, als Sie den IAM-Identity-Center-Benutzer erstellt haben.

Hilfe bei der Anmeldung mit einem IAM Identity Center-Benutzer finden Sie [im AWS-Anmeldung Benutzerhandbuch unter Anmeldung beim AWS Access-Portal](#).

Weiteren Benutzern Zugriff zuweisen

1. Erstellen Sie im IAM-Identity-Center einen Berechtigungssatz, der den bewährten Vorgehensweisen für die Anwendung von geringsten Berechtigungen folgt.

Anweisungen hierzu finden Sie unter [Berechtigungssatz erstellen](#) im AWS IAM Identity Center Benutzerhandbuch.

2. Weisen Sie Benutzer einer Gruppe zu und weisen Sie der Gruppe dann Single Sign-On-Zugriff zu.

Eine genaue Anleitung finden Sie unter [Gruppen hinzufügen](#) im AWS IAM Identity Center Benutzerhandbuch.

Erste Schritte mit MediaTailor

Für den Einstieg MediaTailor können Sie zwischen zwei Tutorials wählen: eines zur Einrichtung der Anzeigeneinfügung und eines zur Kanalmontage. Das Tutorial zur Anzeigeneinfügung führt Sie durch das Senden einer Wiedergabeanfrage, MediaTailor um personalisierte Anzeigen in Ihren Content-Stream zu integrieren. In der Anleitung zur Kanalzusammenstellung erfahren Sie, wie Sie den Stream Ihres Kanals, einschließlich der personalisierten Anzeigen, mithilfe einer Wiedergabe-URL direkt in einem Browser ansehen können.

Themen

- [Erste Schritte mit dem Einfügen MediaTailor von Anzeigen](#)
- [Erste Schritte mit der MediaTailor Kanalmontage](#)

Erste Schritte mit dem Einfügen MediaTailor von Anzeigen

Für die Verwendung benötigen Sie die Berechtigungen AWS-Konto und AWS Elemental MediaTailor, um auf MediaTailor Konfigurationen zuzugreifen, sie anzusehen und zu bearbeiten. Weitere Informationen hierzu finden Sie unter [Einrichten AWS Elemental MediaTailor](#).

Im Tutorial "Erste Schritte" erfahren Sie, wie Sie die folgenden Aufgaben ausführen:

- Bereiten Sie Ihre HLS- oder DASH-Inhaltsstreams vor
- Konfigurieren Sie eine ADS-Vorlagen-URL (Ad Decision Server)
- Erstellen Sie eine MediaTailor Konfiguration, die einen Wiedergabeendpunkt enthält
- Verwenden Sie Ihren Player oder Ihr Content Delivery Network (CDN), um eine Wiedergabeanforderung zu stellen MediaTailor

Wenn du fertig bist, kannst du eine Wiedergabeanfrage MediaTailor für personalisierte Werbeinhalte in deinem Stream senden.

Themen

- [Voraussetzungen](#)
- [Schritt 1: Zugriff AWS Elemental MediaTailor](#)
- [Schritt 2: Bereiten Sie einen Stream vor](#)
- [Schritt 3: Konfigurieren Sie die ADS-Anforderungs-URL und die Abfrageparameter](#)

- [Schritt 4: Erstellen einer Konfiguration](#)
- [Schritt 5: Testen Sie die Konfiguration](#)
- [Schritt 6: Senden Sie die Wiedergabeanforderung an AWS Elemental MediaTailor](#)
- [Schritt 7 \(optional\): AWS Elemental MediaTailor Aktivität überwachen](#)
- [Schritt 8: Bereinigen](#)

Voraussetzungen

Bevor Sie beginnen, sollten Sie sicherstellen, dass Sie die in [Einrichten AWS Elemental MediaTailor](#) beschriebenen Schritte ausgeführt haben.

Schritt 1: Zugriff AWS Elemental MediaTailor

Melden Sie sich mit Ihren IAM-Anmeldeinformationen unter an der MediaTailor Konsole an **`https://console.aws.amazon.com/mediatailor/home`**.

Schritt 2: Bereiten Sie einen Stream vor

Konfigurieren Sie Ihren Ursprungs-Server so, dass mit ihm Manifeste für HLS oder DASH erstellt werden, die mit AWS Elemental MediaTailor kompatibel sind.

Bereiten Sie einen HLS-Stream vor

HLS-Manifeste müssen die folgenden Anforderungen erfüllen:

- Manifeste müssen über das öffentliche Internet zugänglich sein.
- Die Manifeste müssen live oder Video-on-Demand (VOD) sein.
- Manifeste müssen einen EXT-X-VERSION von 3 oder höher haben.
- Für Live-Inhalte müssen Manifeste Marker zur Abgrenzung von Ad-Avails enthalten. Dies ist für On-Demand-Inhalte optional, bei denen stattdessen VMAP-Timeoffsets verwendet werden können.

In der Manifestdatei müssen sich Werbeplätze mit den folgenden Markern befinden:

- `#EXT-X-CUE-OUT` / `#EXT-X-CUE-IN` (häufiger) mit einer Dauer wie im folgenden Beispiel gezeigt.

```
#EXT-X-CUE-OUT:60.00
#EXT-X-CUE-IN
```

- **#EXT-X-DATERANGE** (weniger gebräuchlicher) mit Angaben zur Dauer wie im folgenden Beispiel dargestellt.

```
#EXT-X-DATERANGE:ID="",START-DATE="",DURATION=30.000,SCTE35-OUT=0xF
#EXT-X-DATERANGE:ID="",START-DATE="",DURATION=30.000,SCTE35-OUT=0xF
```

Alle für **#EXT-X-DATERANGE** angezeigten Felder sind erforderlich.

Von der Art und Weise, wie Sie die Werbe-Marker im Manifest konfigurieren, ist abhängig, ob Werbung in einen Stream eingefügt wird oder ob andere Fragmente im Stream durch sie ersetzt werden. Weitere Informationen finden Sie unter [the section called “Das Verhalten bei der Insertion von Anzeigen verstehen”](#).

- HLS-Mastermanifeste müssen der HLS-Spezifikation entsprechen, die unter [HTTP-Live-Streaming: Master-Playlist-Tags](#) dokumentiert ist. Insbesondere muss **#EXT-X-STREAM-INF** die Felder **RESOLUTION**, **BANDWIDTH** und **CODEC** enthalten.

Nachdem Sie den Stream konfiguriert haben, beachten Sie das Präfix der Ursprungs-URL von Inhalten für das Master-Manifest. Sie benötigen es zum Erstellen der Konfiguration in AWS Elemental MediaTailor an späterer Stelle in diesem Tutorial.

Bereiten Sie einen DASH-Stream vor

DASH-Manifeste müssen die folgenden Anforderungen erfüllen:

- Manifeste müssen über das öffentliche Internet zugänglich sein.
- Die Manifeste müssen live oder Video-on-Demand (VOD) sein.
- Manifeste müssen Ereignisse als Ad-Avails markieren und dazu entweder Splice-Insert-Marker oder Zeitsignal-Marker verwenden. Sie können die Werbemarker im transparenten XML- oder im base64-kodierten binären Format angeben. Für das Einfügen von Spleißverbindungen muss der out-of-network Indikator aktiviert sein. Bei Zeitsignal-Markierungen muss es sich bei der Segmentierungstyp-ID, die sich innerhalb der Segmentierungs-UPID befindet, um einen Cue-Out-Wert handeln, der von erkannt wird. AWS Elemental MediaTailor Der Ad-Avail beginnt am Anfang des Ereignisses und dauert über die Ereignisdauer an, sofern angegeben, oder bis das nächste Ereignis beginnt.

Das folgende Beispiel zeigt ein Ereignis, das mittels Splice-Insert-Marker als Ad-Avail gekennzeichnet ist. Die Dauer dieses Ad-Avail entspricht der Ereignisdauer.

```

<Period start="PT444806.040S" id="123586" duration="PT15.000S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="1350000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="180832"
tier="4095">
        <scte35:SpliceInsert spliceEventId="4026531855"
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
          <scte35:Program><scte35:SpliceTime ptsTime="5672624400"/></
scte35:Program>
            <scte35:BreakDuration autoReturn="true" duration="1350000"/>
          </scte35:SpliceInsert>
        </scte35:SpliceInfoSection>
      </Event>
    </EventStream>
    <AdaptationSet mimeType="video/mp4"
      ...
    </AdaptationSet>
  </Period>

```

- Ad-Avails müssen die gleichen AdaptationSet- und Representation-Einstellungen wie die Streams der Inhalte besitzen. AWS Elemental MediaTailor verwendet diese Einstellungen zur Transcodierung der Werbung entsprechend dem Stream der Inhalte, um einen reibungslosen Wechsel zwischen ihnen zu ermöglichen.

Nachdem Sie den Stream konfiguriert haben, beachten Sie das Präfix der Ursprungs-URL von Inhalten für das DASH-Manifest. Sie benötigen sie, um die Konfiguration später in AWS Elemental MediaTailor diesem Tutorial zu erstellen.

Schritt 3: Konfigurieren Sie die ADS-Anforderungs-URL und die Abfrageparameter

Um zu bestimmen, welche Abfrageparameter der ADS benötigt, erstellen Sie eine Werbe-Tag-URL vom ADS. Diese URL fungiert als Vorlage für Anforderungen an den ADS und enthält Folgendes:

- Statische Werte
- Werte, die von AWS Elemental MediaTailor (bezeichnet mit `session` oder `avail` Abfrageparametern) generiert wurden

- Von Playern generierte Werte, die von der Client-Anwendung abgerufen wurden (gekennzeichnet durch `player_params`.-Abfrageparameter)

Example Werbe-Tag-URL von einem ADS

```
https://my.ads.com/ad?  
output=vast&content_id=12345678&playerSession=[session.id]&cust_params=[player_params.cust_params]
```

Wobei gilt:

- `output` und `content_id` sind statische Werte
- `playerSession= [session.id]` ist ein dynamischer Wert, der von bereitgestellt wird. AWS Elemental MediaTailor Der Wert von `[session.id]` ändert sich mit jeder Player-Sitzung und ergibt eine jeweils andere URL für die VAST-Anforderung für jede Sitzung.
- `cust_params` sind vom Player bereitgestellte dynamische Werte

Die Master-Manifest-Anforderung von dem Player muss Schlüssel-Wert-Paare bereitstellen, die den `player_params`.-Abfrageparametern in der ADS-Anforderungs-URL entsprechen. Weitere Informationen über das Konfigurieren von Schlüssel-Wert-Paaren in der Anforderung an AWS Elemental MediaTailor finden Sie unter [Verwendung dynamischer Anzeigenvariablen in MediaTailor](#).

Geben Sie die konfigurierte „Vorlage“-URL ein, wenn Sie das Ursprungs-Mapping in, in erstellen.
server/ADS MediaTailor [Schritt 4: Erstellen einer Konfiguration](#)

Testen

Sie können eine statische VAST-Antwort von Ihrem ADS für Testzwecke verwenden. Im Idealfall gibt die VAST-Antwort eine MP4 Wiedergabe in Mezzanine-Qualität zurück, die transkodiert werden kann. AWS Elemental MediaTailor Wenn die ADS-Antwort mehrere Wiedergabeverionen enthält, wird die Wiedergabeverion mit der höchsten Qualität und Auflösung MediaTailor MP4 ausgewählt und an den Transcoder gesendet.

Schritt 4: Erstellen einer Konfiguration

Die AWS Elemental MediaTailor Konfiguration enthält Zuordnungsinformationen für den Ursprungsserver und ADS.

So erstellen Sie eine Konfiguration (Konsole)

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie auf der Seite Configurations (Konfigurationen) die Option Create configuration (Konfiguration erstellen).
3. Geben Sie im Bereich Configuration (Konfiguration) unten auf der Seite für Configuration name (Konfigurationsname) einen eindeutigen Namen zur Beschreibung der Konfiguration ein. Der Name ist der primäre Kennung für die Konfiguration. Die maximale zulässige Länge beträgt 512 Zeichen.
4. Geben Sie für Video content source (Quelle der Videoinhalte) das URL-Präfix für das HLS-Master-Manifest oder DASH-Manifest für diesen Stream abzüglich der Komponenten-ID ein. Beispiel: Bei der Master-Manifest-URL `http://origin-server.com/a/master.m3u8` würden Sie `http://origin-server.com/a/` eingeben. Alternativ können Sie ein kürzeres Präfix, wie z. B. `http://origin-server.com/`, eingeben, Sie müssen dann aber `/a/` in die Komponenten-ID der Player-Anforderung für Inhalte einschließen. Die maximale Länge beträgt 512 Zeichen.

Note

Wenn der Ursprungs-Server Ihrer Inhalte HTTPS verwendet, muss sein Zertifikat von einer bekannten Zertifizierungsstelle stammen. (Es darf kein selbstsigniertes Zertifikat sein.) Andernfalls kann AWS Elemental MediaTailor keine Verbindung zum Ursprung des Inhalts hergestellt werden und es können keine Manifeste als Antwort auf Spieleranfragen bereitgestellt werden.

5. Geben Sie für Ad Decision Server die URL für Ihre ADS ein. Dies ist entweder die unter [Schritt 3: Konfigurieren Sie die ADS-Anforderungs-URL und die Abfrageparameter](#) beschriebene URL mit Variablen oder die statische VAST-URL, die Sie für Testzwecke verwenden. Die maximale Länge beträgt 25.000 Zeichen.

Note

Wenn der ADS HTTPS verwendet, muss sein Zertifikat von einer bekannten Zertifizierungsstelle stammen. (Es darf kein selbstsigniertes Zertifikat sein.) Das Gleiche gilt für Mezzanine-Anzeigen, die vom ADS URLs zurückgegeben werden. Andernfalls

kann MediaTailor keine Werbespots vom Ursprungs-Server der Inhalte abrufen und in den Manifesten zusammenfügen.

6. (Optional, je nach Bedarf für DASH) Wählen Sie für Standort die Option DEAKTIVIERT, wenn Sie CDN-Routing-Regeln für den Zugriff auf MediaTailor Manifeste eingerichtet haben und entweder clientseitige Berichte verwenden oder Ihre Player Sticky-HTTP-Weiterleitungen unterstützen.

Weitere Informationen über die Funktion Location (Speicherort) finden Sie unter [the section called “Standort-Funktion”](#).

7. (Optional) Wenn Ihr Ursprungs-Server DASH-Manifeste mit einzelnen Zeiträumen erstellt, wählen Sie DASH mpd manifest origin typ (DASH mpd-Manifest-Ursprungstyp) und danach SINGLE_PERIOD aus. MediaTailor Behandelt DASH-Manifeste standardmäßig als Manifeste mit mehreren Perioden. Weitere Informationen finden Sie unter [the section called “Integration einer MPEG-DASH-Quelle”](#).
8. Wählen Sie Create configuration (Konfiguration erstellen).

AWS Elemental MediaTailor zeigt die neue Konfiguration auf der Konfigurationsseite an.

Schritt 5: Testen Sie die Konfiguration

Nachdem Sie die Konfiguration gespeichert haben, testen Sie den Stream mithilfe einer URL im entsprechenden Format für Ihr Streaming-Protokoll:

- Beispiel: HLS

```
playback-endpoint/v1/master/hashed-account-id/origin-id/master.m3u8
```

- Beispiel: DASH

```
playback-endpoint/v1/dash/hashed-account-id/origin-id/manifest.mpd
```

Wobei gilt:

- playback-endpoint ist der eindeutige Wiedergabe-Endpunkt, den AWS Elemental MediaTailor beim Erstellen der Konfiguration generiert hat.

Beispiel

```
https://777788889999.mediatailor.us-east-1.amazonaws.com
```

- `hashed-account-id` ist deine AWS-Konto ID.

Beispiel

```
777788889999
```

- `origin-id` ist der Name, den Sie beim Erstellen der Konfiguration gewählt haben.

Beispiel

```
myOrigin
```

- `master.m3u8` oder `manifest.mpd` ist der Name des Manifests aus dem Test-Stream samt seiner Dateierweiterung. Definieren Sie diese so, dass Sie ein vollständig identifiziertes Manifest erhalten, wenn Sie diese an die Quelle der Videoinhalte anhängen, die Sie unter [the section called “Schritt 4: Erstellen einer Konfiguration”](#) konfiguriert haben.

Unter Verwendung der Werte aus den vorangegangenen Beispielen ergeben sich die vollständigen URLs Werte wie folgt.

- Beispiel: HLS

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/master/  
AKIAIOSFODNN7EXAMPLE/myOrigin/master.m3u8
```

- Beispiel: DASH

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/dash/  
AKIAIOSFODNN7EXAMPLE/myOrigin/manifest.mpd
```

Sie können den Stream mit einer der folgenden Methoden testen.

- Geben Sie wie im vorherigen Beispiel dargestellt die URL in einem eigenständigen Player ein.

- Testen Sie den Stream in Ihrer eigenen Player-Umgebung.

Schritt 6: Senden Sie die Wiedergabeanforderung an AWS Elemental MediaTailor

Konfigurieren Sie den Downstream-Player oder das CDN zum Senden von Wiedergabeanforderungen an den von AWS Elemental MediaTailor bereitgestellten Wiedergabe-Endpunkt der Konfiguration. Alle vom Player definierten dynamischen Variablen, die Sie in der ADS-Anforderungs-URL in [Schritt 3: Konfigurieren Sie die ADS-Anforderungs-URL und die Abfrageparameter](#) verwendet haben, müssen in der Manifest-Anforderung vom Player definiert sein.

Example

Angenommen, Ihre Vorlagen-ADS-URL lautet wie folgt.

```
https://my.ads.com/ad?  
output=vast&content_id=12345678&playerSession=[session.id]&cust_params=[player_params.cust_params]
```

Definieren Sie dann `[player_params.cust_params]` in der Player-Anfrage, indem Sie dem Schlüssel-Wert-Paar das Präfix voranstellen. `ads.` AWS Elemental MediaTailor übergibt Parameter, denen kein Präfix vorangestellt ist `ads.`, an den Ursprungsserver statt an das ADS.

Die Player-Anforderungs-URL ist eine Variante der folgenden HLS- und DASH-Beispiele.

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/master/  
AKIAIOSFODNN7EXAMPLE/myOrigin/master.m3u8?ads.cust_params=viewerinfo
```

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/dash/AKIAIOSFODNN7EXAMPLE/  
myOrigin/manifest.mpd?ads.cust_params=viewerinfo
```

Wenn AWS Elemental MediaTailor die Spieleranfrage erhält, definiert er die Spielervariablen auf der Grundlage der Informationen in der Anfrage. Die resultierende ADS-Anforderungs-URL ist eine Variante hiervon.

```
https://my.ads.com/ad?  
output=vast&content_id=12345678&playerSession=<filled_in_session_id>&cust_params=viewerinfo
```

Weitere Informationen über das Konfigurieren von Schlüssel-Wert-Paaren zur Übergabe an den ADS finden Sie unter [Verwendung dynamischer Anzeigenvariablen in MediaTailor](#).

Schritt 7 (optional): AWS Elemental MediaTailor Aktivität überwachen

Verwenden Sie Amazon CloudWatch und Amazon CloudWatch Logs, um AWS Elemental MediaTailor Aktivitäten nachzuverfolgen, z. B. die Anzahl der Anfragen, Fehler und ausgefüllten Werbeanzeigen.

Wenn Sie zum ersten Mal verwenden CloudWatch AWS Elemental MediaTailor, erstellen Sie eine AWS Identity and Access Management (IAM) -Rolle, um die Kommunikation zwischen den Services zu ermöglichen.

Um den AWS Elemental MediaTailor Zugriff auf CloudWatch (Konsole) zu ermöglichen

1. Öffnen Sie unter <https://console.aws.amazon.com/iam/> die IAM-Konsole.
2. Klicken Sie im Navigationsbereich der IAM-Konsole auf Rollen, und wählen Sie dann Rolle erstellen.
3. Wählen Sie den AWS-Konto Rollentyp „Anderer“.
4. Geben Sie als Konto-ID Ihre AWS-Konto ID ein.
5. Wählen Sie Require external ID (Externe ID fordern) und geben Sie **midas** ein. Mit dieser Option wird automatisch eine Bedingung zur Vertrauensrichtlinie hinzugefügt, die es dem Service nur dann ermöglicht, die Rolle anzunehmen, wenn in der Anforderung die richtige `sts:ExternalID` enthalten ist.
6. Wählen Sie Weiter: Berechtigungen aus.
7. Fügen Sie eine Berechtigungsrichtlinie hinzu, in der die Aktionen angegeben werden, die von dieser Rolle ausgeführt werden dürfen. Wählen Sie eine der folgenden Optionen aus und wählen Sie dann Next: Review (Nächster Schritt: Prüfen):
 - CloudWatchLogsFullAccessum vollen Zugriff auf Amazon CloudWatch Logs zu gewähren
 - CloudWatchFullAccessum vollen Zugriff auf Amazon zu gewähren CloudWatch
8. Geben Sie für Role name (Rollenname) den Namen **MediaTailorLogger** ein und klicken Sie auf Create role (Rolle erstellen).
9. Wählen Sie auf der Seite Roles (Rollen) die von Ihnen soeben erstellte Rolle aus.
10. Bearbeiten Sie die Vertrauensstellung, um den Prinzipal zu aktualisieren:
 1. Wählen Sie auf der Seite Summary (Übersicht) der Rolle die Registerkarte Trust relationship (Vertrauensstellung) aus.
 2. Wählen Sie Vertrauensstellung bearbeiten aus.

3. Ändern Sie im Richtliniendokument den Prinzipal auf den AWS Elemental MediaTailor - Service. Sie sollte wie folgt aussehen.

```
"Principal": {  
  "Service": "mediatailor.amazonaws.com"  
},
```

Die gesamte Richtlinie sollte folgendermaßen lauten.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Principal": {  
        "Service": "mediatailor.amazonaws.com"  
      },  
      "Action": "sts:AssumeRole",  
      "Condition": {  
        "StringEquals": {  
          "sts:ExternalId": "Midas"  
        }  
      }  
    }  
  ]  
}
```

4. Wählen Sie Update Trust Policy (Trust Policy aktualisieren).

Schritt 8: Bereinigen

Um unerwünschte Gebühren zu vermeiden, löschen Sie alle unnötigen Konfigurationen.

So löschen Sie eine Konfiguration (Konsole)

1. Führen Sie auf der Seite AWS Elemental MediaTailor Konfigurationen einen der folgenden Schritte aus:
 - Wählen Sie für die Konfiguration, die Sie löschen möchten, einen Namen für Configuration name (Konfigurationsname) aus.

- Aktivieren Sie in der Spalte Configuration name (Konfigurationsname) das Optionsfeld neben dem Namen und wählen Sie dann Delete (Löschen).
2. Geben Sie im Bestätigungsdialogfeld Delete configuration (Konfiguration löschen) den Text **Delete** ein und wählen Sie erneut Delete (Löschen) aus.

AWS Elemental MediaTailor entfernt die Konfiguration.

Erste Schritte mit der MediaTailor Kanalmontage

Im Tutorial "Erste Schritte" erfahren Sie, wie Sie die folgenden Aufgaben ausführen:

- Erstellen Sie einen Quellspeicherort und fügen Sie diesem Quellinhalt hinzu
- Erstelle einen Kanal
- Erstelle eine Programmliste, um die Inhalte deines Kanals nach einem bestimmten Zeitplan abzuspielen
- Fügen Sie mithilfe AWS Elemental MediaTailor der Anzeigeneinfügung personalisierte Anzeigen zum Kanal-Stream hinzu

Wenn du fertig bist, kannst du einen Browser öffnen, die Wiedergabe-URL für deinen Kanal eingeben und dir den Stream deines Kanals ansehen, der personalisierte Werbung enthält.

Dieses Tutorial führt dich durch die grundlegenden Schritte, um mit der MediaTailor Kanalmontage zu beginnen. Weiterführende Informationen finden Sie unter [Wird verwendet AWS Elemental MediaTailor , um lineare zusammengestellte Streams zu erstellen](#).

Geschätzte Kosten

- Die Gebühr für einen aktiven Kanal beträgt 0,10\$ pro Stunde. Für Kanäle, die inaktiv sind, werden Ihnen keine Gebühren berechnet.

Themen

- [Voraussetzungen](#)
- [Schritt 1: Erstellen Sie einen Quellspeicherort](#)
- [Schritt 2: Fügen Sie VOD-Quellen zu Ihrem Quellstandort hinzu](#)
- [Schritt 3: Erstellen Sie einen Kanal](#)

- [Schritt 4: Füge Programme zum Zeitplan deines Kanals hinzu](#)
- [Schritt 5 \(optional\): Verwenden Sie diese Option MediaTailor , um personalisierte Anzeigen in Ihren Stream einzufügen](#)
- [Schritt 6: Starte deinen Kanal](#)
- [Schritt 7: Teste deinen Kanal](#)
- [Schritt 8: Bereinigen](#)

Voraussetzungen

Bevor Sie mit diesem Tutorial beginnen, müssen Sie die folgenden Anforderungen erfüllen:

- Vergewissern Sie sich, dass Sie die Schritte unter abgeschlossen haben [Einrichten AWS Elemental MediaTailor](#).
- Sie müssen über Ressourcen sowohl für VOD-Quellinhalte als auch für Ad Slate verfügen. Sie müssen den Pfad zu den Manifesten für die Assets kennen.

Note

Wenn Sie ABR (Automated Adaptive Bitrate) oder die Kodierung pro Titel verwenden, müssen Sie Ihre Inhalte so codieren, dass alle Varianten dieselbe Länge haben und dieselbe Anzahl an untergeordneten Titeln haben. Wir empfehlen, eine Kodierungsvorlage mit einer Mindestsegmentlänge von einer Sekunde zu verwenden.

Schritt 1: Erstellen Sie einen Quellspeicherort

Ein Quellspeicherort stellt den Ursprungsserver dar, auf dem Ihre Inhalte gespeichert sind. Dabei kann es sich um Amazon S3, einen Standard-Webserver, ein Content Delivery Network (CDN) oder einen Verpackungshersteller handeln, z. B. AWS Elemental MediaPackage

MediaTailor ruft die Inhaltsmanifeste von Ihrem Quellverzeichnis ab und verwendet sie, um ein Live-Slide-Manifestfenster zusammenzustellen, das auf die zugrunde liegenden Inhaltssegmente verweist.

Gehen Sie wie folgt vor, um einen Quellspeicherort zu erstellen.

Um einen Quellspeicherort zu erstellen

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.

2. Wählen Sie im Navigationsbereich Channel-Assembly > Source Locations aus.
3. Wählen Sie in der Navigationsleiste die Option Quellspeicherort erstellen aus.
4. Geben Sie unter Konfiguration des Quellverzeichnisses eine Kennung und den Speicherort Ihres Quellinhalts ein:
 - Name: Eine Kennung für Ihren Quellstandort, z. B. my-origin.
 - Basis-URL: Die Basis-URL des Ursprungsservers, auf dem Ihre Inhalte gehostet werden, z. B. <https://111111111111.cloudfront.net>. Die URL muss in einem Standard-HTTP-URL-Format mit dem Präfix <http://> oder <https://> vorliegen.
5. Wählen Sie Quellspeicherort erstellen aus.

Schritt 2: Fügen Sie VOD-Quellen zu Ihrem Quellstandort hinzu

Nachdem du einen oder mehrere Quellorte für deinen Kanal definiert hast, kannst du eine oder mehrere VOD-Quellen hinzufügen. Jede VOD-Quelle steht für einen einzelnen Inhalt, z. B. einen einzelnen Film, eine Folge einer TV-Sendung oder einen Highlight-Clip.

Sie müssen mindestens eine Paketkonfiguration für Ihre VOD-Quelle erstellen. Jede Paketkonfiguration enthält das Paketformat und die Manifesteinstellungen für Ihre VOD-Quellen. Anschließend fügen Sie Ihrem Kanal Ihre Paketkonfigurationen hinzu, um Ausgaben zu erstellen.

Sie können mehrere Paketkonfigurationen verwenden, um unterschiedliche Kanalausgänge zu erstellen. Wenn Ihre VOD-Quelle beispielsweise sowohl als HLS als auch als DASH verpackt ist, können Sie zwei Paketkonfigurationen für jedes Format erstellen. Sie können dann die Quellgruppen der Paketkonfiguration verwenden, um zwei Kanalausgänge zu erstellen: einen für HLS, einen für DASH.

Um VOD-Quellen hinzuzufügen und Paketkonfigurationen zu erstellen

1. Öffnen Sie die MediaTailor Konsole unter: <https://console.aws.amazon.com/mediatailor/>
2. Wählen Sie im Navigationsbereich Channel-Assembly > Source Locations aus.
3. Wählen Sie im Bereich Quellverzeichnisse den Quellspeicherort aus, den Sie in dem [Um einen Quellspeicherort zu erstellen](#) Verfahren erstellt haben.
4. Wählen Sie VOD-Quelle hinzufügen aus.
5. Geben Sie unter VOD-Quelldetails einen Namen für Ihre VOD-Quelle ein, z. B. my-example-video

6. **source-group-name** Geben Sie unter Paketkonfigurationen > Informationen zur Paketkonfiguration ein:

 Note

Die Paketkonfigurationen Ihrer Quelle müssen alle dieselbe Dauer haben, wie im Manifest der Quelle festgelegt. Und alle Quellen innerhalb einer Paketkonfiguration müssen dieselbe Anzahl von untergeordneten Streams haben. Um diese Anforderungen zu erfüllen, empfehlen wir Ihnen, eine Kodierungsvorlage für Ihre Assets zu verwenden. Wir empfehlen, eine Kodierungsvorlage mit einer Mindestsegmentlänge von einer Sekunde zu verwenden. MediaTailor unterstützt kein Streaming pro Titel oder automatisiertes Streaming mit adaptiver Bitrate (ABR), da diese Kodierungsmethoden gegen diese Anforderungen verstoßen.

- Quellgruppe: Geben Sie einen Quellgruppennamen ein, der diese Paketkonfiguration beschreibt, z. B. HLS-4k. Notieren Sie sich diesen Namen. Sie werden ihn angeben, wenn Sie die Ausgabe Ihres Kanals erstellen. Weitere Informationen finden Sie unter [Verwende Quellgruppen mit den Ausgaben deines Kanals](#).
 - Typ: Wählen Sie das Paketformat für diese Konfiguration aus. MediaTailor unterstützt HLS und DASH.
 - Relativer Pfad: Der relative Pfad von der Basis-HTTP-URL des Quellspeicherorts zum Manifest. Zum Beispiel /my/path/index.m3u8.
7. Wählen Sie Add source (Quelle hinzufügen) aus.
8. Wiederholen Sie die Schritte 4-7 in diesem Verfahren, um die VOD-Quelle für Ihre Anzeigenliste hinzuzufügen.

Schritt 3: Erstellen Sie einen Kanal

Ein Kanal fasst Ihre Quellen zu einem linearen Live-Stream zusammen. Jeder Kanal enthält einen oder mehrere Ausgänge, die den Paketkonfigurationen Ihrer VOD-Quelle entsprechen.

Zuerst erstellen Sie einen Kanal und fügen dann Ihre VOD-Quellen dem Zeitplan des Kanals hinzu, indem Sie Programme erstellen.

So erstellen Sie einen Channel

1. Öffnen Sie die MediaTailor Konsole unter. <https://console.aws.amazon.com/mediatailor/>
2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie in der Navigationsleiste die Option Kanal erstellen aus.
4. Gib unter Kanaldetails Details zu deinem Kanal ein:
 - Name: Gib einen Namen für deinen Kanal ein.
 - Wiedergabemodus: Legt fest, welche Art von Programmübergängen zulässig sind und was mit einem Programm passiert, wenn es beendet ist. Verwenden Sie den Standard-Loop-Modus.
5. Wählen Sie Weiter aus.
6. Definieren Sie unter Ausgabedetails die Einstellungen für diese Ausgabe:
 - Manifestname: Geben Sie einen Manifestnamen ein, z. **index**B. MediaTailor hängt die Formaterweiterung an, z. B. .m3u8 für HLS.

Note

Sie müssen pro Kanalausgabe einen eindeutigen Manifestnamen eingeben.

- Formattyp: Wählen Sie das Streaming-Format für den Kanal aus. DASH und HLS werden unterstützt. Wählen Sie das Format, das der Paketkonfiguration entspricht, in [Schritt 1: Erstellen Sie einen Quellspeicherort](#) der Sie es erstellt haben.
 - Quellgruppe: Geben Sie den Namen der Quellgruppe ein, in der Sie erstellt haben [Schritt 1: Erstellen Sie einen Quellspeicherort](#).
7. Geben Sie unter Manifesteinstellungen zusätzliche Informationen zu Ihren Manifesteinstellungen ein:
 - Manifestfenster (Sekunden): Das Zeitfenster (in Sekunden), das in jedem Manifest enthalten ist. Der Mindestwert ist 30 Sekunden und der Höchstwert ist 3600 Sekunden.
 8. Wählen Sie Weiter aus.
 9. Wählen Sie unter Kanalrichtlinie die Option Kanalrichtlinie nicht anhängen aus. Diese Option beschränkt die Wiedergabe nur auf Personen, die Zugriff auf Ihre AWS-Konto Anmeldeinformationen haben.
 10. Wählen Sie Weiter aus.
 11. Überprüfen Sie Ihre Einstellungen im Bereich Überprüfen und erstellen.

12. Wählen Sie Create channel (Channel erstellen).

Note

Kanäle werden im Status „Gestoppt“ erstellt. Dein Kanal wird erst aktiv sein, wenn du ihn startest.

Schritt 4: Füge Programme zum Zeitplan deines Kanals hinzu

Jetzt, wo du einen Kanal hast, fügst du Programme zum Zeitplan des Kanals hinzu. Jedes Programm enthält eine VOD-Quelle von einem Quellort in Ihrem Konto. Der Kanalplan bestimmt die Reihenfolge, in der Ihre Programme im Stream des Kanals abgespielt werden.

Jedes Programm kann eine oder mehrere Werbeunterbrechungen haben. Sie fügen eine Werbeunterbrechung ein, indem Sie eine VOD-Quelle angeben, die als Werbetafel verwendet werden soll. Die Dauer der Werbeunterbrechung wird durch die Dauer der Werbeunterbrechung bestimmt. Sie können optional einen serverseitigen Anzeigeneinfügungsserver verwenden, z. B. für die MediaTailor Anzeigeneinfügung, um Ihre Werbeunterbrechungen zu personalisieren.

Um Programme zum Zeitplan deines Kanals hinzuzufügen

1. Öffne die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie im Bereich Kanäle den Kanal aus, den Sie in dem [Schritt 3: Erstellen Sie einen Kanal](#) Verfahren erstellt haben.
4. Geben Sie unter Programmdetails Details zu Ihrem Programm ein:
 - Name: Dies ist der Name des Programms, das zum Zeitplan deines Kanals hinzugefügt werden soll.
 - Name des Quellstandorts: Wähle „Bestehenden Quellstandort auswählen“ und wähle den Quellstandort [Schritt 1: Erstellen Sie einen Quellspeicherort](#) aus, den du im Drop-down-Menü „Quellstandort auswählen“ erstellt hast.
 - Name der VOD-Quelle: Wählen Sie „Eine vorhandene VOD-Quelle auswählen“ und wählen Sie die VOD-Quelle aus, die Sie zuvor in diesem Tutorial erstellt haben.
5. Definieren Sie unter Wiedergabekonfiguration, wie und wann ein Programm in den Zeitplan eines Kanals eingefügt wird:

- **Übergangstyp:** Dieser Wert ist auf Relativ festgelegt. Der relative Übergangstyp gibt an, dass dieses Programm relativ zu anderen Programmen in der Programmliste auftritt.
- **Relative Position:** Wenn dies das erste Programm im Zeitplan Ihres Kanals ist, können Sie diese Einstellung überspringen. Wenn es nicht das erste Programm im Programm Ihres Kanals ist, wähle aus, an welcher Stelle in der Programmliste das Programm angehängt werden soll. Du kannst „Vor der Sendung“ oder „Nach der Sendung“ auswählen.
- **Relatives Programm:** Wenn dies das erste Programm in Ihrem Zeitplan ist, können Sie diese Einstellung überspringen. Wenn es nicht das erste Programm im Programm Ihres Kanals ist, wähle Bestehendes Programm verwenden und wähle den Programmnamen aus, in dem du es erstellt hast [Um Programme zum Zeitplan deines Kanals hinzuzufügen](#).

6.

Wähle „Werbeunterbrechung hinzufügen“ aus. Konfigurieren Sie unter Werbeunterbrechungen die Einstellungen für die Werbeunterbrechung:

- **Name des Speicherorts der Slate-Quelle:** Wählen Sie einen vorhandenen Quellspeicherort auswählen und wählen Sie den Quellspeicherort aus, an dem Ihr Slate gespeichert ist, den Sie zuvor in diesem Tutorial erstellt haben.
- **Name der VOD-Quelle:** Wählen Sie „Eine vorhandene VOD-Quelle auswählen“ und wählen Sie die VOD-Quelle aus, die Sie für Slate verwenden und die Sie zuvor in diesem Tutorial hinzugefügt haben. Die Dauer des Slate bestimmt die Dauer der Werbepause.
- **Für Offset in Millisekunden:** Dieser Wert bestimmt die Startzeit der Werbeunterbrechung in Millisekunden als Offset im Verhältnis zum Beginn des Programms. Geben Sie einen Wert ein, der kürzer als die Dauer der VOD-Quelle ist und der an einer Segmentgrenze für alle Titel innerhalb der VOD-Quelle des Programms (alle Audio-, Video- und Untertitelspuren) ausgerichtet ist. Andernfalls wird die Werbeunterbrechung übersprungen. Wenn Sie beispielsweise 0 eingeben, entsteht eine Pre-Roll-Werbeunterbrechung, die vor Programmbeginn abgespielt wird. Hinweis:.

7. Wählen Sie Programm hinzufügen.

Weitere Informationen zu Programmen finden Sie unter [Configuring ad breaks for your program](#).

Weiterführende Informationen zur Verwendung von Anzeigen mit Ihrem linearen Stream finden Sie unter [Optionale Konfigurationseinstellungen](#).

Schritt 5 (optional): Verwenden Sie diese Option MediaTailor , um personalisierte Anzeigen in Ihren Stream einzufügen

Sie haben jetzt einen Kanal mit Programmen. Wenn Sie möchten, können MediaTailor Sie personalisierte Anzeigen in die Werbeunterbrechungen Ihrer Programme im Stream des Kanals einfügen.

Voraussetzungen

Bevor Sie fortfahren können, müssen Sie die folgenden Anforderungen erfüllen:

- Sie müssen über einen Ad Decision Server (ADS) verfügen.
- Sie müssen im Rahmen des [Ein Programm zum Zeitplan eines Kanals hinzufügen](#) Verfahrens die Einstellungen für Werbeunterbrechungen konfiguriert haben.

So fügst du personalisierte Werbung zum Stream deines Kanals hinzu MediaTailor

1. Öffne die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Konfigurationen aus.
3. Geben Sie unter Erforderliche Einstellungen die grundlegenden erforderlichen Informationen zu Ihrer Konfiguration ein:
 - Name: Der Name Ihrer Konfiguration.
 - Inhaltsquelle: Gib die Wiedergabe-URL aus der Ausgabe deines Kanals ein, abzüglich des Dateinamens und der Erweiterung. Weiterführende Informationen zur MediaTailor Konfiguration finden Sie unter [Erforderliche Einstellungen](#).
 - Entscheidungsserver hinzufügen: Geben Sie die URL für Ihr ADS ein.
4. Sie können optional die Konfigurationsalias, die Personalisierungsdetails und die erweiterten Einstellungen konfigurieren. Informationen zu diesen Einstellungen finden Sie unter [Optionale Konfigurationseinstellungen](#)
5. Wählen Sie in der Navigationsleiste Konfiguration erstellen aus.

Weiterführende Informationen zur Verwendung der MediaTailor Anzeigeneinfügung finden Sie unter [AWS Elemental MediaTailor Zum Einfügen von Anzeigen verwenden](#).

Schritt 6: Starte deinen Kanal

Du hast jetzt einen Kanal. Bevor Sie jedoch auf den Stream des Kanals zugreifen können, müssen Sie Ihren Kanal starten. Wenn Sie versuchen, auf einen Kanal zuzugreifen, bevor er aktiv ist, wird ein 4xx HTTP-Fehlercode MediaTailor zurückgegeben.

Starte deinen Kanal

1. Öffne die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie in der Navigationsleiste „Start“ aus.

Schritt 7: Teste deinen Kanal

Um zu überprüfen, ob dein Kanal korrekt funktioniert, öffne einen Webbrowser und gib die URL aus der Ausgabe deines Kanals ein. Du solltest den Stream deines Kanals sehen.

In einigen Fällen müssen Sie möglicherweise den Cache löschen, um das erwartete Verhalten zu sehen.

Schritt 8: Bereinigen

Wenn Sie mit dem Kanal fertig sind, den Sie für dieses Tutorial erstellt haben, sollten Sie ihn löschen, indem Sie ihn löschen.

Sobald sich der Kanalstatus auf Gestoppt ändert, fallen keine Gebühren mehr für diesen Kanal an. Wenn du deinen Kanal für später behalten möchtest, ohne dass Gebühren anfallen, kannst du den Kanal jetzt beenden und später wieder starten.

Um deinen Kanal zu löschen

1. Öffne die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie den Kanal aus, den Sie löschen möchten.
4. Wenn dein Kanal läuft, wähle im Drop-down-Menü Aktionen die Option Stopp aus. Du musst deinen Kanal beenden, bevor du ihn löschen kannst.
5. Wenn dein Kanal gestoppt ist, wähle im Drop-down-Menü „Aktionen“ die Option Löschen aus.

AWS Elemental MediaTailor Zum Einfügen von Anzeigen verwenden

Eine Konfiguration ist ein Objekt, mit dem Sie in AWS Elemental MediaTailor interagieren. Die Konfiguration enthält die Zuordnungsinformationen für den Ursprungsserver und den Ad Decision Server (ADS). Sie können auch eine Standardwiedergabe definieren MediaTailor , die verwendet wird, wenn eine Anzeige nicht verfügbar ist oder nicht das gesamte Anzeigenangebot ausfüllt.

Wenn Sie ein Content Distribution Network (CDN) mit verwenden MediaTailor, müssen Sie die Verhaltensregeln im CDN einrichten, bevor Sie CDN-Informationen zur Konfiguration hinzufügen. Weitere Informationen zum Einrichten Ihres CDN finden Sie unter [Ein CDN integrieren](#).

Themen

- [Unterstützte Audio- und Videocodecs](#)
- [Das Verhalten bei der AWS Elemental MediaTailor Anzeigeneinblendung verstehen](#)
- [Grundlegendes zur AWS Elemental MediaTailor servergesteuerten Anzeigeneinfügung](#)
- [Anforderungen für Adserver-Integrationen mit AWS Elemental MediaTailor](#)
- [Mit Wiedergabekonfigurationen arbeiten AWS Elemental MediaTailor](#)
- [Integration einer Inhaltsquelle für die MediaTailor Anzeigeneinfügung](#)
- [Integration AWS Elemental MediaTailor mit Google Ad Manager](#)
- [Verwendung eines CDN zur Optimierung der Anzeigenpersonalisierung und Inhaltsbereitstellung](#)
- [Anpassen des Verhaltens von Werbeunterbrechungen mit Anzeigenunterdrückung](#)
- [Stoßstangen einsetzen](#)
- [Pre-Roll-Anzeigen einfügen](#)
- [Schiefer wird eingefügt](#)
- [Anzeigen vorab abrufen](#)
- [Verwenden von vorkonditionierten Anzeigen mit AWS Elemental MediaTailor](#)
- [Verwendung dynamischer Anzeigenvariablen in MediaTailor](#)
- [Übergabe von AWS Elemental MediaTailor Sitzungsinitialisierungsparametern an das Manifest](#)
- [Daten zur Berichterstattung und Nachverfolgung](#)
- [Overlay-Anzeigen](#)
- [Werbe-ID-Dekoration](#)

Unterstützte Audio- und Videocodecs

MediaTailor unterstützt die folgenden Codecs.

- Audio-Codecs: mp4a, ac-3, und ec-3
- Videocodecs: h.264 (AVC), h.265 (HEVC), av01 (AV1)

Das Verhalten bei der AWS Elemental MediaTailor Anzeigeneinblendung verstehen

AWS Elemental MediaTailor fügt Anzeigen in Live- oder Video-on-Demand-Inhalte (VOD) ein, indem Anzeigen entweder ersetzt oder in das Original-Manifest eingefügt werden. Ob Anzeigen eingefügt oder ersetzt werden, hängt davon ab, wie die Werbeunterbrechungen im Ursprungsmanifest konfiguriert sind und ob es sich um VOD- oder Live-Inhalte handelt. Bei einer Werbeunterbrechung handelt es sich um den Zeitraum während der Programmgestaltung, in dem Werbespots gezeigt werden. Bei der Verfügbarkeit von Werbeanzeigen handelt es sich um die spezifischen Einheiten der Werbezeit innerhalb einer Werbepause, die mit Werbung gefüllt werden kann.

- Bei der Anzeigenersetzung werden Inhaltssegmente durch Anzeigen MediaTailor ersetzt.
- MediaTailor Fügt beim Einfügen von Anzeigen Anzeigeninhalte dort ein, wo es keine Segmente gibt.

Informationen dazu, wie MediaTailor Anzeigen in Live- und VOD-Inhalte eingebunden werden, erhalten Sie unter dem entsprechenden Thema.

Themen

- [Verhalten beim Zusammenfügen von Anzeigen bei VOD](#)
- [Verhalten beim Zusammenfügen von Live-Anzeigen](#)

Verhalten beim Zusammenfügen von Anzeigen bei VOD

MediaTailor fügt Anzeigen in VOD-Inhalte ein oder ersetzt sie, je nachdem, wie die Anzeigenmarkierungen im Ursprungsmanifest konfiguriert sind und ob der Ad Decision Server (ADS) VMAP-Antworten sendet.

Die nachstehenden Abschnitte gehen auf das Werbeverhalten je nach Marker-Konfiguration ein.

Wenn Werbemarkierungen vorhanden sind

AWS Elemental MediaTailor fügt Anzeigen ein, bei denen SCTE-35-Anzeigenmarkierungen vorhanden sind, im Ursprungsmanifest vorhanden sind. Werbemarkierungen mit dem EXT-X-CUE-OUT Wert 0 Dauer weisen auf das Einfügen einer Anzeige hin.

Richtlinien für HLS-Werbemarkierungen

Beachten Sie diese Richtlinien für die SCTE-Signalisierung nach dem Rollvorgang und beim Ad-Pod:

Pre-Roll-Anzeigen

Für HLS-Post-Rolls müssen dem letzten Inhaltssegment CUE-OUT/IN-Markierungen vorangestellt werden. Der Grund hierfür ist, dass laut HLS-Spezifikation Tag-Decoratoren explizit vor einem Segment deklariert werden müssen.

Betrachten wir beispielsweise die folgende Deklaration.

```
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXTINF:4.000,
Videocontent.ts
#EXT-X-ENDLIST
```

AWS Elemental MediaTailor fügt eine Postroll-Anzeige wie die folgende ein.

```
#EXTINF:4.000,
Videocontent.ts
#EXT-X-DISCONTINUITY
#EXTINF:3.0,
Adsegment1.ts
#EXTINF:3.0,
Adsegment2.ts
#EXTINF:1.0,
Adsegment3.ts
#EXT-X-ENDLIST
```

Example 2: Pods hinzufügen

CUE-OUT/INTags müssen explizit an ein Segment angehängt werden. Es ist nicht möglich, mehrere CUE-OUT/IN-Tags nacheinander zu verwenden, um Werbe-Pod-Verhalten nachzuahmen.

Die folgende Deklaration ist beispielsweise eine gültige Verwendung von CUE-OUT/IN, um einen Werbe-Pod darzustellen.

```
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXTINF:4.000,
Somecontent1.ts
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXTINF:4.000,
Somecontent2.ts
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXTINF:4.000,
Videocontent.ts
```

Die obige Deklaration führt zu einer Ausgabe wie der folgenden.

```
Ad 1
Somecontent.ts
Ad 2
Somecontent2.ts
Videocontent.ts
Post-Roll Ad 3
```

Die folgende Deklaration ist ungültig.

```
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXT-X-CUE-OUT: 0
#EXT-X-CUE-IN
#EXTINF:4.000,
Videocontent.ts
```

Wenn keine Werbemarkierungen vorhanden sind

Werbemarkierungen sind die empfohlene Methode, um Werbeunterbrechungen in einem Manifest zu signalisieren. Anzeigenmarkierungen sind jedoch nicht erforderlich. Wenn das Manifest keine

Werbemarkierungen für DASH oder HLS enthält, MediaTailor ruft das ADS einmal auf und erstellt auf der Grundlage der Antwort Werbeunterbrechungen:

- Wenn das ADS eine VAST-Antwort sendet, werden zu Beginn des Manifests alle Anzeigen aus der Antwort in einen Werbeblock MediaTailor eingefügt. Dies wird als Pre-Roll (Vorlauf) bezeichnet.
- Wenn das ADS eine VMAP-Antwort sendet, MediaTailor verwendet es die Zeitversätze für Werbeunterbrechungen, um Pausen zu erstellen und sie zu den angegebenen Zeiten (Pre-Roll, Mid-Roll oder Post-Roll) in das gesamte Manifest einzufügen. MediaTailor verwendet alle Anzeigen aus den einzelnen Werbeunterbrechungen in der VMAP-Antwort für jede Werbeunterbrechung im Manifest.

 Note

Wenn ein Segment einen Einfügepunkt mit VMAP für VOD-Inhalten überlappt, rundet MediaTailor auf den nächsten Einfügepunkt ab.

 Tip

Wenn Sie Mid-Roll-Werbepausen erstellen möchten, der ADS aber nicht VMAP unterstützt, muss das Manifest Werbemarker enthalten. MediaTailor fügt (wie in den nachstehenden Abschnitten beschrieben) an den Markern Werbung ein.

Verhalten beim Zusammenfügen von Live-Anzeigen

Führt in Livestreams AWS Elemental MediaTailor immer eine Anzeigenersetzung durch, wobei die Gesamtzeit zwischen den Anzeigenmarkierungen so genau wie möglich beibehalten wird. Wenn Anzeigenmarkierungen das DURATION Attribut enthalten, bestimmt MediaTailor anhand des Werts die Dauer der Werbeunterbrechung. In Live-Workflows muss jeder CUE-OUT Indikator eine Dauer oder einen entsprechenden CUE-IN Indikator haben.

MediaTailor führt einen Werbersatz für HLS- und DASH-Live-Inhalte durch. Informationen zur MediaTailor Berechnung der Platzierung und des Zeitpunkts von Werbeunterbrechungen finden Sie unter [the section called “Werbe-Marker”](#) und [the section called “Werbe-Marker”](#)

Auswahl und Ersetzung von Werbeanzeigen

AWS Elemental MediaTailor umfasst Anzeigen aus der VAST-Antwort des Anzeigen-Entscheidungservers (ADS) wie folgt:

- Wenn eine Dauer angegeben ist, wird eine Gruppe von Anzeigen MediaTailor ausgewählt, die in die Dauer passen, und schließt sie ein.
- Wenn keine Dauer angegeben ist, werden so viele Anzeigen wie möglich MediaTailor abgespielt, bis eine Werbemarkierung erscheint, die auf eine Rückkehr zum Hauptinhalt hinweist.

AWS Elemental MediaTailor hält sich beim Ersetzen von Live-Anzeigen an die folgenden Richtlinien:

- MediaTailor versucht, komplette Werbeanzeigen abzuspielen, ohne sie auszuschneiden oder zu kürzen.
- Immer wenn Sie MediaTailor auf eine Werbemarkierung stoßen, die auf das Ende der Werbeunterbrechung hinweist, kehrt sie zum zugrunde liegenden Inhalt zurück. Dies kann bedeuten, dass eine laufende Werbung gekürzt wird.
- Kehrt am Ende der Laufzeit zum zugrunde liegenden Inhalt MediaTailor zurück.
- Wenn MediaTailor für die Dauer einer Werbepause keine Werbeanzeigen mehr verfügbar sind, wird entweder der Slate abgespielt, sofern ein solcher konfiguriert ist, oder die Wiedergabe des zugrunde liegenden Inhaltsstreams fortgesetzt. Dies geschieht normalerweise, wenn nicht genügend transkodierte Werbung vorhanden ist, um die Dauer der Werbepause auszufüllen.

Tip

Mit der Konfigurationseinstellung für eine Personalisierungsschwelle können Sie den Grenzwert für die nicht ausgefüllte Werbezeit in einer Pause festlegen. Weitere Informationen finden Sie in der [PlaybackConfiguration Referenz](#).

Beispiele

- Wenn die Dauer der Werbepause auf 70 Sekunden festgelegt ist und die ADS-Antwort zwei 40-Sekunden-Anzeigen enthält, wird eine der 40-Sekunden-Anzeigen AWS Elemental MediaTailor abgespielt. In der übrigen Zeit wechselt es zum konfigurierten Pausenfüller oder zu den zugrunde

liegenden Inhalten. Wenn MediaTailor zu irgendeinem Zeitpunkt während dieses Prozesses einen Cue-in-Indikator feststellt, wechselt es sofort zu den zugrunde liegenden Inhalten.

- Wenn die Dauer der Werbeunterbrechung auf 30 Sekunden festgelegt ist und die kürzeste Anzeige in der ADS-Antwort 40 Sekunden lang ist, werden keine Anzeigen MediaTailor abgespielt. Wenn ein Ad Slate konfiguriert ist, wird dieser 30 Sekunden lang MediaTailor abgespielt oder bis ein Cue-In-Indikator angezeigt wird. Andernfalls wird der zugrunde liegende MediaTailor Inhalt abgespielt.

Grundlegendes zur AWS Elemental MediaTailor servergesteuerten Anzeigeneinfügung

Die serverseitige Anzeigeneinfügung (HLS-Interstitials) ist eine Alternative zur serverseitigen Anzeigeneinfügung. Anstatt Anzeigen direkt in Medien-Playlists einzufügen, wird auf Anzeigen als separate primäre Playlist verwiesen. Dies ermöglicht schnellere Videostartzeiten und geringere Latenzen bei der Manifestation.

Für Informationen zur Verwendung der servergesteuerten Anzeigeneinfügung mit wählen Sie MediaTailor das entsprechende Thema aus.

Themen

- [In der Wiedergabekonfiguration aktivieren](#)
- [Erstellen Sie eine servergeführte Sitzung](#)

In der Wiedergabekonfiguration aktivieren

Damit Spieler die servergesteuerte Anzeigeneinfügung verwenden können, müssen Sie `PLAYER_SELECT` in der MediaTailor Wiedergabekonfiguration die Einstellung `Insertion Mode` auf einstellen. Auf diese Weise können Spieler bei der Sitzungsinitialisierung zwischen gestickter oder gesteuerter Anzeigeneinfügung wählen.

Erstellen Sie eine servergeführte Sitzung

Wählen Sie beim Erstellen von Wiedergabesitzungen den geführten Modus. Wie das geht, hängt davon ab, ob Ihre Spieler implizite oder explizite Sessions verwenden.

Implizit erstellte servergeführte Sitzungen

An die übergeordnete `aws.insertionMode=GUIDED` HLS-Manifestanforderung anhängen.

Beispiel:

```
playback-endpoint/v1/master/hashed-account-id/origin-id/index.m3u8?  
aws.insertionMode=GUIDED
```

Wobei gilt:

- `playback-endpoint` ist der eindeutige Wiedergabe-Endpunkt, den AWS Elemental MediaTailor beim Erstellen der Konfiguration generiert hat.

Beispiel

```
https://777788889999.mediatailor.us-east-1.amazonaws.com
```

- `hashed-account-id` ist deine AWS-Konto ID.

Beispiel

- `origin-id` ist der Name, den Sie beim Erstellen der Konfiguration gewählt haben.

Beispiel

```
myOrigin
```

- `index.m3u8` oder ist der Name des Manifests aus dem Teststream plus dessen Dateierweiterung. Definieren Sie diese so, dass Sie ein vollständig identifiziertes Manifest erhalten, wenn Sie diese an die Quelle der Videoinhalte anhängen, die Sie unter [the section called “Schritt 4: Erstellen einer Konfiguration”](#) konfiguriert haben.

Unter Verwendung der Werte aus den vorangegangenen Beispielen ergeben sich die vollständigen URLs Werte wie folgt.

- Beispiel:

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/master/  
AKIAIOSFODNN7EXAMPLE/myOrigin/index.m3u8?aws.insertionMode=GUIDED
```

Explizit erstellte servergeführte Sitzungen

Fügen Sie `insertionMode=GUIDED` die JSON-Metadaten hinzu, die der Player im HTTP POST an den Präfixendpunkt für die Sitzungsinitialisierung der MediaTailor Konfiguration sendet.

Das folgende Beispiel zeigt die Struktur der JSON-Metadaten:

```
{
  # other keys, e.g. "adsParams"
  "insertionMode": "GUIDED"      # this can be either GUIDED or STITCHED
}
```

Mit diesen Initialisierungsmetadaten verwendet die Wiedergabesitzung die serverseitige Anzeigeneinfügung.

Anforderungen für Adserver-Integrationen mit AWS Elemental MediaTailor

Um Ihren Ad-Server zu integrieren AWS Elemental MediaTailor, muss Ihr Ad-Server XML senden, das den IAB-Spezifikationen für die unterstützten Versionen von VAST und VMAP entspricht. Sie können mithilfe einer öffentlichen VAST-Validierung sicherstellen, dass Ihre Tags ein gültiges Format aufweisen.

AWS Elemental MediaTailor unterstützt VAST- und VMAP-Antworten von Anzeigenentscheidungsservern. AWS Elemental MediaTailor unterstützt auch die Weiterleitung von VPAID-Metadaten über unsere clientseitige Berichts-API für die clientseitige Anzeigeneinblendung. Weitere Informationen zur clientseitigen Berichterstellung finden Sie unter [Kundenseitiges Ad-Tracking](#).

MediaTailor unterstützt die folgenden Versionen von VAST, VMAP und VPAID:

- [Bis zu VAST 4.3](#)

MediaTailor akzeptiert Antwortversionen bis VAST 4.3, aber einige erweiterte Funktionen ab VAST 4.0 werden nicht unterstützt.

- [VMAP 1.0](#)
- [VPAID 2.0](#)

RIESIGE Anforderungen

Die VAST-Antwort Ihres Webservers muss IAB-konforme TrackingEvents-Elemente und Standard-Ereignistypen, wie z. B. `impression`, enthalten. Wenn Sie nicht standardmäßige Tracking-Ereignisse einbeziehen, AWS Elemental MediaTailor lehnt die VAST-Antwort ab und stellt keine Werbeanzeige für den Erfolg bereit.

Mit VAST 3.0 wurde die Unterstützung von Werbe-Pods eingeführt. Dabei handelt es sich um die Bereitstellung einer Reihe von sequenziellen linearen Werbespots. Wenn eine bestimmte Anzeige in einem Ad-Pod nicht verfügbar ist, AWS Elemental MediaTailor wird im Interaktionsprotokoll des ADS ein Fehler protokolliert. CloudWatch Anschließend versucht es, die nächste Werbung im Pod einzufügen. Auf diese Weise werden die Anzeigen im Pod MediaTailor wiederholt, bis eine gefunden wird, die verwendet werden kann.

Targeting

Um bestimmte Spieler für Ihre Anzeigen anzusprechen, können Sie Vorlagen für Ihre Anzeigen-Tags und URLs erstellen. Weitere Informationen finden Sie unter [Verwendung dynamischer Anzeigenvariablen in MediaTailor](#).

AWS Elemental MediaTailor leitet die Werte `user-agent` und `x-forwarded-for` Header des Players weiter, wenn er die VAST-Anfrage des Anzeigenservers sendet und wenn er serverseitige Tracking-Aufrufe tätigt. Stellen Sie sicher, dass Ihre Werbeserver diese Header verarbeiten können. Alternativ können Sie `[session.user_agent]` oder `[session.client_ip]` verwenden und diese Werte in Abfragezeichenfolgen für das Werbe-Tag und die Werbe-URL übergeben. Weitere Informationen finden Sie unter [Verwenden von Sitzungsvariablen](#).

Ad-Aufrufe

AWS Elemental MediaTailor ruft Ihre VAST-Anzeigen-URL wie in Ihrer Konfiguration definiert auf. Sie ersetzt beim Anzeigenaufruf alle spieler- oder sitzungsspezifischen Parameter. MediaTailor folgt bis zu sieben Ebenen von VAST-Wrappern und leitet in der VAST-Antwort um. Führt in Live-Streaming-Szenarien zu Beginn der Werbeanzeige für verbundene Spieler gleichzeitig Werbeanrufe durch. MediaTailor In der Praxis können sich diese Werbeaufrufe aufgrund von Jitter über mehrere Sekunden erstrecken. Stellen Sie sicher, dass Ihr Anzeigenserver die Anzahl gleichzeitiger Verbindungen verarbeiten kann, die für diese Art von Anrufen erforderlich sind. MediaTailor unterstützt das Vorabrufen von VAST-Antworten für Live-Workflows. Weitere Informationen finden Sie unter [Anzeigen vorab abrufen](#).

Kreativer Umgang

Beim AWS Elemental MediaTailor Empfang der ADS VAST-Antwort wird für jedes Creative die höchste Bitrate MediaFile für die Transcodierung ermittelt und diese als Quelle verwendet. Diese Datei wird an den on-the-fly Transcoder gesendet, wo sie in Wiedergabeversionen umgewandelt wird, die den Bitraten und Auflösungen des Players entsprechen. Um optimale Ergebnisse zu erzielen, sollten Sie sicherstellen, dass es sich bei Ihrer Mediendatei mit der höchsten Bitrate um ein qualitativ hochwertiges MP4 Asset mit gültigen Manifest-Voreinstellungen handelt. Wenn die Manifest-Voreinstellungen nicht gültig sind, schlagen die transcodierten Aufträge fehl und wird keine Werbung angezeigt. Zu den ungültigen Voreinstellungen gehören beispielsweise nicht unterstützte Eingabedateiformate und bestimmte Wiedergabespezifikationen wie ProRes die Auflösung 855X481.

Eine Liste der unterstützten Formate für Mediendateieingaben finden Sie in der MP4Zeile [Unterstützte Eingabeformate](#) im Benutzerhandbuch.AWS Elemental MediaConvert

Kreative Indizierung

AWS Elemental MediaTailor Indiziert jedes Creative eindeutig anhand des Werts des im Element angegebenen id Attributs. <Creative> Wenn die ID eines Kreativen nicht angegeben ist, wird die URL der Mediendatei für den Index MediaTailor verwendet.

Die folgende Beispieldeklaration zeigt die Werbemittel-ID.

```
<Creatives>
  <Creative id="57859154776" sequence="1">
```

Wenn Sie Ihr eigenes Creative definieren IDs, verwenden Sie für jedes Creative eine neue, eindeutige ID. Kreatives Material nicht wiederverwenden IDs. AWS Elemental MediaTailor speichert kreative Inhalte zur wiederholten Verwendung und findet sie anhand ihrer indizierten ID. Wenn ein neues Werbemittel eingeht, überprüft der Service zunächst den Index auf seine ID. Wenn die ID vorhanden ist, verwendet MediaTailor die gespeicherten Inhalte, ohne die eingehenden Inhalte erneut zu verarbeiten. Wenn Sie eine Creative-ID wiederverwenden MediaTailor , die ältere, gespeicherte Anzeige verwenden und Ihre neue Anzeige nicht wiedergeben.

VAST-Erweiterungen, die von Ad-Serving-Partnern bereitgestellt werden

Um Kollisionen mit Werbematerial zu vermeiden IDs, können Sie Erweiterungen verwenden, die von Werbepartnern für die VAST-Antwort bereitgestellt werden. MediaTailor unterstützt Erweiterungen von SpringServe, Publica und. FreeWheel Wenn Sie VAST-Erweiterungsüberschreibungen aktivieren, wird die standardmäßige Creative-ID durch den Erweiterungswert MediaTailor ersetzt.

Um diese Funktion zu aktivieren, [reichen Sie ein AWS Support-Ticket ein](#) und beantragen Sie, dass auf VAST-Erweiterungen basierende IDs Werbemittel aktiviert werden. Nehmen Sie die folgenden Informationen in das Support-Ticket auf:

- AWS Region
- AWS Konto-ID
- MediaTailor Namen der Wiedergabekonfigurationen

Um zu überprüfen, ob Creative, die auf VAST-Erweiterungen basieren, in Ihrem Konto aktiviert IDs sind, empfehlen wir Ihnen, auch die RAW_ADS_RESPONSE Protokollierung für eine Staging- oder Testkonfiguration für die Wiedergabe zu aktivieren. Mithilfe der Protokollierung können Sie sich die ursprüngliche VAST-Antwort ansehen, die das ADS erhält, und sich vergewissern, dass die richtigen Werbemittel IDs verwendet wurden.

VPAID-Anforderungen

Mithilfe von VPAID können Herausgeber äußerst interaktive Werbevideos aufnehmen und Sichtbarkeitsmetriken für ihre monetarisierte Streams bereitstellen. Informationen zu VPAID finden Sie in der [VPAID-Spezifikation](#).

AWS Elemental MediaTailor unterstützt eine Mischung aus MP4 linearen server-side-stitched VAST-Anzeigen und interaktiven client-side-inserted VPAID-Kreativen in derselben Anzeigenversion. Dabei wird die Reihenfolge beibehalten, in der sie in der VAST-Antwort erscheinen. MediaTailor folgt VPAID-Weiterleitungen durch maximal sieben Ebenen von Wrappern. Die Antwort des clientseitigen Berichts enthält die entpackten VPAID-Metadaten.

Beachten Sie zur Verwendung von VPAID die folgenden Richtlinien:

- Konfigurieren Sie einen MP4 Slate für Ihre VPAID-Creatives. AWS Elemental MediaTailor füllt die VPAID-Anzeigenfelder mit Ihrem konfigurierten Slate und stellt dem Kundenplayer VPAID-Anzeigenmetadaten zur Verfügung, die er zur Schaltung der interaktiven Anzeigen verwenden kann. Wenn bei Erscheinen einer VPAID-Werbung kein konfigurierter Pausenfüller vorhanden ist, stellt MediaTailor die Werbe-Metadaten wie üblich über die clientseitige Berichtserstellung bereit. Außerdem wird eine Fehlermeldung CloudWatch bezüglich des fehlenden Slate protokolliert. Weitere Informationen erhalten Sie unter [Schiefer wird eingefügt](#) und [Eine Konfiguration erstellen](#).
- Verwenden Sie die clientseitige Berichterstattung. AWS Elemental MediaTailor unterstützt VPAID über unsere clientseitige Berichts-API. Weitere Informationen finden Sie unter [Kundenseitiges Ad-Tracking](#).

Es ist theoretisch auch möglich, den serverseitigen Standard-Berichterstellungsmodus mit VPAID zu verwenden. Wenn Sie die serverseitige Berichterstellung verwenden, verlieren Sie jedoch alle Informationen über das Vorhandensein der VPAID-Werbung und der zugehörigen Metadaten, da diese Informationen nur über die clientseitige API verfügbar sind.

- Stellen Sie in Live-Szenarien sicher, dass Ihre mit gekennzeichneten Anzeigenverkäufe lang genug sind **EXT-X-CUE-OUT: Duration**, um jegliche Nutzerinteraktivität auf VPAID zu ermöglichen. Wenn das VAST-XML beispielsweise eine VPAID-Anzeige mit einer Länge von 30 Sekunden angibt, sollten Sie erwägen, Ihre Anzeigenverfügbarkeit auf mehr als 30 Sekunden zu konfigurieren. Diese zusätzliche Zeit gibt Nutzern mehr Möglichkeiten, mit der Anzeige zu interagieren. Wenn Sie keine Zeit hinzufügen, könnten Sie die VPAID-Metadaten verlieren, da die verbleibende Dauer der Anzeigenverfügbarkeit nicht lang genug ist, um die VPAID-Anzeige aufzunehmen.

Mit Wiedergabekonfigurationen arbeiten AWS Elemental MediaTailor

In diesem Abschnitt werden die wichtigsten Aufgaben zur Verwaltung von MediaTailor Wiedergabekonfigurationen behandelt. Sie erfahren, wie Sie eine neue Konfiguration erstellen, um Inhaltsstreams einzurichten und den Zugriff für Wiedergabegeräte zu ermöglichen, Details einer vorhandenen Konfiguration anzuzeigen, eine Konfiguration zu bearbeiten, um Einstellungen wie Ursprungsserver und Anzeigenentscheidungsserver zu aktualisieren, und eine Konfiguration zu löschen, die nicht mehr benötigt wird.

Themen

- [Eine Konfiguration erstellen](#)
- [Konfiguration anzeigen](#)
- [Konfiguration bearbeiten](#)
- [Eine Konfiguration löschen](#)

Eine Konfiguration erstellen

In diesem Thema wird gezeigt, wie Sie eine Konfiguration erstellen, um mit dem Empfang von Inhaltsstreams zu beginnen. Außerdem wird gezeigt, wie ein Zugriffspunkt für nachgeschaltete Wiedergabegeräte bereitgestellt wird, um Inhalte anzufordern.

Sie können die AWS Elemental MediaTailor Konsole, AWS Command Line Interface (AWS CLI) > oder die MediaTailor API verwenden, um eine Konfiguration zu erstellen. Informationen zum Erstellen einer Konfiguration über die MediaTailor API AWS CLI oder finden Sie in der [AWS Elemental MediaTailor API-Referenz](#).

Wenn Sie eine Konfiguration erstellen, geben Sie keine vertraulichen Identifikationsinformationen in Freiformfelder wie das Feld für den Konfigurationsnamen ein. Zu den identifizierenden Informationen können Dinge wie Kundenkontonummern gehören. Verwenden Sie außerdem keine identifizierenden Informationen, wenn Sie in der MediaTailor Konsole, der REST-API AWS CLI, dem oder arbeiten AWS SDKs. Alle Daten, die Sie eingeben, werden MediaTailor möglicherweise für die Aufnahme in Diagnoseprotokolle oder Amazon CloudWatch Events aufgenommen.

So fügen Sie eine Konfiguration hinzu (Konsole)

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie auf der Seite Configurations (Konfigurationen) die Option Create configuration (Konfiguration erstellen).
3. Füllen Sie die Konfigurationsfelder und zusätzlichen Konfigurationsfelder wie in den folgenden Themen beschrieben aus:
 - [Erforderliche Einstellungen](#)
 - [Optionale Konfigurationseinstellungen](#)
4. Wählen Sie Create configuration (Konfiguration erstellen).

AWS Elemental MediaTailor zeigt die neue Konfiguration in der Tabelle auf der Konfigurationsseite an.

5. (Empfohlen) Richten Sie ein CDN AWS Elemental MediaTailor für Manifest- und Berichtsanforderungen ein. Sie können die Konfigurationswiedergabe URLs für das CDN-Setup verwenden. Weitere Informationen zum Einrichten eines CDN für Manifest- und Berichterstellungs-Anforderungen finden Sie unter [Ein CDN integrieren](#).

Erforderliche Einstellungen

Wenn Sie eine Konfiguration erstellen, müssen Sie die folgenden erforderlichen Einstellungen angeben.

Name

Geben Sie einen eindeutigen Namen ein, der die Konfiguration beschreibt. Der Name ist der primäre Kennung für die Konfiguration. Die maximale zulässige Länge beträgt 512 Zeichen.

Quelle des Inhalts

Geben Sie das URL-Präfix für das Manifest für diesen Stream ein, ohne die Asset-ID. Die maximale Länge beträgt 512 Zeichen.

Das URL-Präfix `http://origin-server.com/a/` ist beispielsweise gültig für die URL des übergeordneten HLS-Manifests `http://origin-server.com/a/main.m3u8` und für die URL des DASH-Manifests von `http://origin-server.com/a/dash.mpd`. Alternativ können Sie ein kürzeres Präfix wie z. B. `http://origin-server.com` eingeben, `/a/` muss dann aber in die Komponenten-ID der Player-Anforderung für Inhalte eingeschlossen werden.

Note

Wenn der Ursprungs-Server Ihrer Inhalte HTTPS verwendet, muss sein Zertifikat von einer bekannten Zertifizierungsstelle stammen. Es kann kein selbstsigniertes Zertifikat sein. Wenn Sie ein selbstsigniertes Zertifikat verwenden, AWS Elemental MediaTailor schlägt die Verbindung zum Inhaltsursprung fehl und Sie können keine Manifeste als Antwort auf Spieleranfragen bereitstellen.

Und Entscheidungsserver

Geben Sie die URL für Ihren Ad Decision Server (ADS) ein. Dies ist entweder die unter [Schritt 3: Konfigurieren Sie die ADS-Anforderungs-URL und die Abfrageparameter](#) beschriebene URL mit Variablen oder die statische VAST-URL, die Sie für Testzwecke verwenden. Die maximale Länge beträgt 25.000 Zeichen.

Note

Wenn der ADS HTTPS verwendet, muss sein Zertifikat von einer bekannten Zertifizierungsstelle stammen. Es kann kein selbstsigniertes Zertifikat sein. Das Gleiche

gilt auch für Mezzanine-Anzeigen, die vom ADS URLs zurückgegeben werden. Wenn Sie ein selbstsigniertes Zertifikat verwenden, AWS Elemental MediaTailor können Sie keine Anzeigen vom Ursprung des Inhalts abrufen und in die Manifeste einbinden.

Optionale Konfigurationseinstellungen

Sie können optional Konfigurationsalias, Personalisierungsdetails und erweiterte Einstellungen in der MediaTailor Konsole, der MediaTailor API oder der AWS Command Line Interface () konfigurieren. AWS CLI

Aliase für die Konfiguration

Im Folgenden finden Sie optionale Konfigurationsalias, die Sie in der MediaTailor Konsole oder mit der MediaTailor API konfigurieren können.

Variable für den Spielerparameter

Fügen Sie für die dynamische Domänenkonfiguration während der Sitzungsinitialisierung eine oder mehrere Player-Parametervariablen hinzu.

Weitere Hinweise zur Verwendung von Player-Parametervariablen zur dynamischen Konfiguration von Domänen finden Sie unter [Verwenden von Domänenvariablen zur Konfiguration mehrerer Inhalts- und Anzeigenquellen](#).

Konfiguration protokollieren

Im Folgenden sind die Einstellungen für die Protokollkonfiguration aufgeführt.

Prozent aktiviert

Legt den Prozentsatz der Sitzungsprotokolle zur Wiedergabekonfiguration fest, der in CloudWatch Protokolle MediaTailor geschrieben wird. Wenn Ihre Wiedergabekonfiguration beispielsweise 1000 Sitzungen umfasst und Sie die Prozentzahl auf 60 gesetzt haben, werden 600 Sitzungsprotokolle in CloudWatch Logs MediaTailor geschrieben.

Wenn Sie diese Option aktivieren, MediaTailor wird automatisch eine dienstbezogene Rolle erstellt, mit der Sie Sitzungsprotokolle in Ihrem CloudWatch Logs-Konto schreiben und verwalten können MediaTailor . Weitere Informationen finden Sie unter [Verwenden von serviceverknüpften Rollen für MediaTailor](#).

Strategien für die Protokollierung

Gibt die Methode an, die zum Sammeln von ausgegebenen Protokollen verwendet wird. MediaTailor Um Protokolle direkt an CloudWatch Logs zu senden, wählen Sie `LEGACY_CLOUDWATCH`. Um Logs an CloudWatch Logs zu senden, das die Logs dann an das Ziel Ihrer Wahl weiterleitet, wählen Sie `VENDED_LOGS`. Unterstützte Ziele sind eine CloudWatch Logs-Protokollgruppe, ein Amazon S3 S3-Bucket und ein Amazon Data Firehose-Stream.

Für verkaufte Protokolle ist eine zusätzliche Einrichtung erforderlich. Informationen zur Einrichtung finden Sie unter [Verwendung von verkauften Protokollen](#).

Opt-in-Ereignisse für das ADS-Interaktionsprotokoll

Gibt an, dass MediaTailor `RAW_ADS_RESPONSE` Protokolle für Sitzungen ausgegeben werden, die mit dieser Konfiguration initialisiert wurden.

Das `RAW_ADS_RESPONSE` Protokollereignis enthält die gesamte VAST- oder VMAP-Antwort des ADS. Daher können die Protokolle umfangreich sein und Ihre Protokollierungskosten erhöhen.

Das ADS-Interaktionsprotokoll schließt Ereignisse aus

Zeigt an, dass die ausgewählten Ereignisse in den Protokollen, die Interaktionen mit dem ADS beschreiben, MediaTailor nicht ausgegeben werden.

Eine Beschreibung der ADS-Protokollereignisse finden Sie unter [ADS-Protokolle](#).

Manifestes Dienstinteraktionsprotokoll schließt Ereignisse aus

Gibt an, dass die ausgewählten Ereignisse in den Protokollen, die Interaktionen mit dem Manifestdienst beschreiben, MediaTailor nicht ausgegeben werden.

Eine Beschreibung der Manifest-Dienstprotokollereignisse finden Sie unter [Manifest-Protokolle](#).

Und Konditionierung

Im Folgenden wird festgelegt, welche Maßnahmen ergriffen MediaTailor werden, um Anzeigen zu konditionieren, bevor sie in einen Inhaltsstream eingefügt werden.

Konditionierung von Streaming-Mediendateien

Legt die Logik fest, nach der entschieden MediaTailor wird, welche Anzeigen zusammengefügt werden sollen.

- Wenn die Konditionierung von Streaming-Mediendateien auf Transkodieren gesetzt ist, werden die Mediendateien bei der progressive Auslieferung MediaTailor transkodiert und in das Manifest zusammengefügt. Wenn nicht genügend Mediendateien für Werbeanzeigen mit progressive Versand zur Verfügung stehen, werden diese Dateien MediaTailor transkodiert und bei Lieferung verwendet. `streaming`
- Wenn die Konditionierung von Streaming-Mediendateien auf „Keine“ gesetzt ist, werden Anzeigen mit MediaTailor `streaming` Übertragungsmediendateien in das Manifest zusammengefügt, ohne sie zu transkodieren. Wenn nicht genügend Werbeanzeigen mit Mediendateien zur `streaming` Auslieferung vorhanden sind, werden diese Dateien MediaTailor transkodiert und bei der Auslieferung verwendet. `progressive`

Einzelheiten zur Personalisierung

Im Folgenden finden Sie Personalisierungsdetails, die Sie in der MediaTailor Konsole oder mit der MediaTailor API konfigurieren können.

Slate-Werbung

Geben Sie die URL für ein qualitativ hochwertiges MP4 Asset ein, das transkodiert und zum Ausfüllen der Zeit verwendet werden soll, die nicht für Werbeanzeigen verwendet wird. AWS Elemental MediaTailor zeigt die Tafel an, um Lücken in Medieninhalten zu schließen. Das Konfigurieren des Pausenfüllers ist für Nicht-VPAID-Konfigurationen optional. Für VPAID müssen Sie ein Slate konfigurieren, das MediaTailor in den Slots für dynamische Anzeigeninhalte vorgesehen ist. Bei dem Slate muss es sich um ein qualitativ hochwertiges MP4 Asset handeln, das sowohl Audio als auch Video enthält. Weitere Informationen finden Sie unter [Schiefer wird eingefügt](#).

Note

Wenn der Server, auf dem der Pausenfüller gehostet wird, HTTPS verwendet, muss sein Zertifikat von einer bekannten Zertifizierungsstelle stammen. Es kann kein selbstsigniertes Zertifikat sein. Wenn Sie ein selbstsigniertes Zertifikat verwenden, AWS Elemental MediaTailor können Sie das Slate aus dem Inhaltsursprung nicht abrufen und in die Manifeste einbinden.

Stoßstange starten

Die URL des Speicherorts des Start-Bumper-Assets. Bumper sind kurze Video- oder Audioclips, die zu Beginn oder am Ende einer Werbepause abgespielt werden. Sie können auf Amazon S3 oder einem anderen Speicherdienst gespeichert werden. Weitere Informationen über Stoßstangen finden Sie unter [Stoßstangen einsetzen](#).

Endstoßstange

Die URL des Asset-Standorts am Ende der Stoßstange. Bumper sind kurze Video- oder Audioclips, die zu Beginn oder am Ende einer Werbepause abgespielt werden. Sie können auf Amazon S3 oder einem anderen Speicherdienst gespeichert werden. Weitere Informationen über Stoßstangen finden Sie unter [Stoßstangen einsetzen](#).

Personalisierungsschwelle

Definiert die maximale Dauer der nicht ausgefüllten Werbezeit (in Sekunden), die in einer Werbepause zulässig ist. Wenn die Dauer der nicht ganz ausgefüllten Werbezeit die Personalisierungsschwelle überschreitet, wird die Personalisierung der Werbepause aufgegeben und der zugrunde liegende Inhalt wird angezeigt. Lautet der Wert der Personalisierungsschwelle beispielsweise 3 und der Pausenfüller einer Werbepause ist z. B. 4 Sekunden lang, wird die Personalisierung der Werbepause aufgegeben und der zugrunde liegende Inhalt wird angezeigt. Diese Funktion gilt für das Ersetzen von Inhalten durch Werbung in Live- und VOD-Streams und nicht für das Einblenden von Werbung, da dies auf einem zugrunde liegenden Inhaltsstream beruht. Weitere Informationen zum Verhalten von Werbeunterbrechungen, einschließlich des Ersetzens und Einfügens von Werbeanzeigen, finden Sie unter [Das Verhalten bei der AWS Elemental MediaTailor Anzeigeneinblendung verstehen](#).

Live-Pre-Roll-Werbeentscheidungsserver

Um Werbung am Anfang eines Live-Streams einzufügen, bevor der Hauptinhalt mit der Wiedergabe beginnt, geben Sie die URL für das Ad-Preroll vom Ad Decision Server (ADS) ein. Dies ist entweder die unter [Schritt 3: Konfigurieren Sie die ADS-Anforderungs-URL und die Abfrageparameter](#) beschriebene URL mit Variablen oder die statische VAST-URL, die Sie für Testzwecke verwenden. Die maximale Länge beträgt 25.000 Zeichen.

Note

Wenn der ADS HTTPS verwendet, muss sein Zertifikat von einer bekannten Zertifizierungsstelle stammen. Es kann kein selbstsigniertes Zertifikat sein. Das Gleiche gilt auch für Mezzanine-Anzeigen, die vom ADS URLs zurückgegeben werden. Wenn Sie

ein selbstsigniertes Zertifikat verwenden, AWS Elemental MediaTailor können Sie keine Anzeigen vom Ursprung des Inhalts abrufen und in die Manifeste einbinden.

Informationen zur Funktionsweise von Pre-Roll finden Sie unter [Pre-Roll-Anzeigen einfügen](#).

Maximal zulässige Live-Pre-Roll-Dauer

Wenn Sie Werbung am Anfang eines Livestreams einfügen, geben Sie die maximal zulässige Dauer für die verfügbaren Pre-Roll-Werbung ein. MediaTailor wird diese Dauer beim Einfügen von Werbung nicht überschreiten. Wenn die Antwort aus dem ADS mehr Anzeigen enthält, als in diese Dauer passen, wird das MediaTailor Angebot mit so vielen Anzeigen wie möglich aufgefüllt, ohne die Dauer zu überschreiten. Weitere Informationen darüber, wie MediaTailor Fills verfügbar sind, finden Sie unter [Verhalten beim Zusammenfügen von Live-Anzeigen](#)

Verwenden Sie den Unterdrückungsmodus

Legt den Modus für die Werbeplatzunterdrückung fest, wird auch als Werbeunterdrückung bezeichnet. Standardmäßig ist die Anzeigenunterdrückung ausgeschaltet und alles wird mit Werbung oder Slate MediaTailor gefüllt. Wenn der Modus auf eingestellt ist `BEHIND_LIVE_EDGE`, ist die Anzeigenunterdrückung aktiv und füllt MediaTailor keine Werbeunterbrechungen, die an oder hinter dem Wert für die Verfügbarkeitsunterdrückung im Manifest-Lookback-Fenster liegen. Wenn der Modus auf eingestellt ist `AFTER_LIVE_EDGE`, ist die Anzeigenunterdrückung aktiv. MediaTailor füllt keine Werbeunterbrechungen auf oder hinter dem Zeitraum der Verfügbarkeitsunterbindung, der sich aus dem Live-Rand plus dem Wert für die Verfügbarkeitsunterbindung plus Pufferzeit zusammensetzt.

Nutzt den Wert für die Unterdrückung

Der Wert für die Avail-Unterdrückung entspricht der Offsetzeit der Live-Kante in. HH:MM:SS MediaTailor Ich werde im Manifest-Lookback-Fenster keine Werbeunterbrechungen an oder hinter diesem Zeitpunkt auffüllen.

Einfügemodus

Der Einfügemodus steuert, ob Spieler das Einfügen von eingefügten oder geführten Anzeigen verwenden können. Die Standardeinstellung zwingt alle Playersitzungen dazu `STITCHED_ONLY`, gestitched (serverseitig) Ad Insertion zu verwenden. Bei der Einstellung InsertionMode auf `PLAYER_SELECT` können Spieler bei der Initialisierung der Sitzung zwischen gestickter oder geführter Anzeigeneinfügung wählen. Die Standardeinstellung für Player, die keinen Einfügemodus angeben, ist „Stitched“.

Erweiterte Einstellungen

Bei den folgenden optionalen Einstellungen handelt es sich um erweiterte Einstellungen. Sie können diese in der MediaTailor Konsole, mit der AWS Command Line Interface (AWS CLI) oder mithilfe der MediaTailor API konfigurieren.

CDN-Inhaltssegmentpräfix

Ermöglicht AWS Elemental MediaTailor die Erstellung von Manifesten mit URLs Ihrem CDN-Pfad für Inhaltssegmente. Bevor Sie diesen Schritt ausführen, richten Sie in Ihrem CDN eine Regel für den Abruf von Segmenten vom Ursprungs-Server ein. Geben Sie für CDN content segment prefix (CDN-Inhaltssegment-Präfix) den CDN-Präfixpfad ein.

Weitere Informationen zur Integration MediaTailor mit einem CDN finden Sie unter [Verwendung eines CDN zur Optimierung der Anzeigenpersonalisierung und Inhaltsbereitstellung](#)

CDN-Webesegmentpräfix

Ermöglicht AWS Elemental MediaTailor die Erstellung von Manifesten mit URLs Ihrem eigenen CDN-Pfad für Anzeigensegmente. Standardmäßig werden MediaTailor Anzeigensegmente aus einer internen CloudFront Amazon-Distribution mit Standard-Cache-Einstellungen bereitgestellt. Bevor Sie das Feld CDN ad segment prefix (CDN-Webesegment-Präfix) ausfüllen können, müssen Sie wie in dem folgenden Beispiel in Ihrem CDN eine Regel für den Abruf von Segmenten von dem folgenden Ursprungs-Server einrichten.

```
https://segments.mediatailor.<region>.amazonaws.com
```

Geben Sie für CDN ad segment prefix (CDN-Webesegmentpräfix) den Namen Ihres CDN-Präfix in der Konfiguration ein.

Weitere Informationen zur Integration MediaTailor mit einem CDN finden Sie unter [Verwendung eines CDN zur Optimierung der Anzeigenpersonalisierung und Inhaltsbereitstellung](#).

DASH-Origin Manifesttyp

Wenn Ihr Ursprungsserver DASH-Manifeste mit einem einzigen Zeitraum produziert, öffnen Sie die Dropdown-Liste und wählen Sie SINGLE_PERIOD. Standardmäßig behandelt MediaTailor DASH-Manifeste wie Manifeste mit mehreren Zeiträumen. Weitere Informationen finden Sie unter [the section called “Integration einer MPEG-DASH-Quelle”](#).

DASH-mpd Speicherort

(Optional, je nach Bedarf für DASH) Der Speicherort der Beschreibung der Medienpräsentation (MPD). Wählen Sie DISABLED für die folgende Situation:

- Sie richten CDN-Routingregeln für den Zugriff auf MediaTailor Manifeste ein.
- Sie verwenden clientseitige Berichterstattung, oder Ihr Player unterstützt Sticky-HTTP-Weiterleitungen.

Weitere Informationen über die Funktion Location (Speicherort) finden Sie unter [the section called “Standort-Funktion”](#).

Transkodieren Sie den Profilnamen

Der Name, der diese Konfiguration einem benutzerdefinierten Transcodierungsprofil zuordnet. Dieser Name überschreibt die Standardeinstellungen für die dynamische Transcodierung von MediaTailor. Füllen Sie dieses Feld nur aus, wenn Sie mithilfe des AWS Supports bereits benutzerdefinierte Profile eingerichtet haben.

Passthrough für Werbemarkierungen

Aktiviert oder deaktiviert für HLS den Passthrough für Werbemarkierungen. Wenn der Passthrough für Werbemarkierungen aktiviert ist, werden MediaTailor EXT-X-CUE-IN, EXT-X-CUE-OUT, und EXT-X-SPLICEPOINT-SCDTE35 Anzeigenmarkierungen vom Ursprungsmanifest zum personalisierten Manifest weitergeleitet. MediaTailor wendet die Werte der Werbemarkierung keine Logik angewendet. Sie werden unverändert vom Ursprungsmanifest an das personalisierte Manifest übergeben. Wenn es beispielsweise 60 im Ursprungsmanifest den Wert 0 EXT-X-CUE-OUT hat, aber keine Werbung geschaltet MediaTailor wird, wird der Wert 0 im personalisierten Manifest nicht geändert.

Konfiguration anzeigen

MediaTailor zeigt zusätzlich zu den Werten, die bei der Erstellung der Konfiguration angegeben wurden, den Namen der Konfiguration, die Wiedergabeendpunkte und den entsprechenden Zugriff URLs an. Gehen Sie wie folgt vor, um eine Konfiguration anzuzeigen.

So zeigen Sie eine Konfiguration an

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.

2. Wählen Sie auf der Seite Configurations (Konfigurationen) unter Configuration name (Konfigurationsname) den Namen der anzuzeigenden Konfiguration aus.

Konfiguration bearbeiten

Sie können eine Konfiguration bearbeiten, um die Zuordnung zwischen dem Ursprungsserver und dem AD Decision Server (ADS) zu aktualisieren oder die Art und Weise der AWS Elemental MediaTailor Interaktion mit einem Content Distribution Network (CDN) zu ändern.

So bearbeiten eine Konfiguration

1. Öffnen Sie die MediaTailor Konsole unter. <https://console.aws.amazon.com/mediatailor/>
2. Wählen Sie auf der Seite Configurations (Konfigurationen) den Namen der Konfiguration aus, die Sie bearbeiten möchten.
3. Klicken Sie auf der Seite mit den Details auf Edit (Bearbeiten) und überarbeiten Sie dann die Konfigurationseinstellungen nach Bedarf. Der Konfigurationsname kann nicht bearbeitet werden. Weitere Informationen über Konfigurationsattribute finden Sie unter [Eine Konfiguration erstellen](#).
4. Wählen Sie Save aus.

Eine Konfiguration löschen

Sie können eine Konfiguration löschen, um ihre Verfügbarkeit für die Wiedergabe aufzuheben.

So löschen Sie eine Konfiguration

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Führen Sie auf der Seite Configurations (Konfigurationen) einen der folgenden Schritte durch:
 - Wählen Sie den Namen der Konfiguration aus, die Sie löschen möchten.
 - Aktivieren Sie in der Spalte Configuration name (Konfigurationsname) das Optionsfeld neben dem Namen und wählen Sie dann Delete (Löschen).
3. Geben Sie im Bestätigungsdiaologfeld Delete (Löschen) den Text **Delete** ein und wählen Sie dann Delete (Löschen) aus.

Integration einer Inhaltsquelle für die MediaTailor Anzeigeneinfügung

Dieses Thema behandelt die Integration verschiedener Arten von Videoinhaltsquellen mit MediaTailor. MediaTailor unterstützt sowohl HLS- als auch DASH-Streaming-Protokolle für Live- und On-Demand-Inhalte. Der Dienst kann das Einfügen oder Ersetzen von Anzeigen während bestimmter Werbepausen vornehmen und stellt spezielle Anforderungen an die Struktur und Formatierung der Eingabevideomanifeste, um diese Funktionen zu aktivieren. In den folgenden Themen finden Sie Einzelheiten zu den Anforderungen an die Eingabequelle und zu den Schritten für die Integration von HLS- und DASH-Inhalten MediaTailor, um personalisierte Werbeerlebnisse zu ermöglichen.

Themen

- [Anforderungen an die Eingabequelle für die MediaTailor Anzeigenschaltung](#)
- [Integration einer HLS-Quelle](#)
- [Integration einer MPEG-DASH-Quelle](#)
- [Sicherung der AWS Elemental MediaTailor ursprünglichen Interaktionen mit SigV4](#)

Anforderungen an die Eingabequelle für die MediaTailor Anzeigenschaltung

Eine Eingabequelle muss die folgenden Anforderungen erfüllen, um damit arbeiten zu können MediaTailor:

- Er muss Apple HLS (HTTP Live Streaming) oder MPEG DASH (Dynamic Adaptive Streaming über HTTP) verwenden.
- Er muss Live-Streaming oder Video-On-Demand (VOD) verwenden.
- Er muss über das öffentliche Internet zugänglich sein und über eine öffentliche IP-Adresse verfügen.
- Sie müssen Werbemarkierungen in einem der im [Tutorial Erste Schritte beim Einfügen von MediaTailor Anzeigen](#) beschriebenen Formate enthalten

Integration einer HLS-Quelle

AWS Elemental MediaTailor unterstützt .m3u8 HLS-Manifeste mit einem Wert EXT-X-VERSION von 3 oder höher für Live-Streaming und Video on Demand (VOD). Wenn MediaTailor auf eine Werbeunterbrechung gestoßen wird, wird je nach Art des Inhalts versucht, eine Anzeige einzufügen

oder zu ersetzen. Wenn nicht genügend Anzeigen für die gesamte Dauer vorhanden sind, wird für den Rest der Werbeunterbrechung der zugrunde liegende Inhaltsstream oder der konfigurierte Slate MediaTailor angezeigt. Weitere Informationen zum Verhalten von HLS-Anzeigen je nach Inhaltstyp finden Sie unter [Das Verhalten bei der AWS Elemental MediaTailor Anzeigeneinblendung verstehen](#).

In den folgenden Abschnitten finden Sie weitere Informationen zum MediaTailor Umgang mit HLS-Manifesten.

Themen

- [Von HLS unterstützte Anzeigenmarkierungen](#)
- [Passthrough für Werbemarkierungen aktivieren](#)
- [Behandlung von HLS-Manifest-Tags](#)
- [Beispiele für HLS-Manifeste](#)

Von HLS unterstützte Anzeigenmarkierungen

AWS Elemental MediaTailor identifiziert und nutzt Grenzen in einem HLS-Manifest, indem das Eingabemanifest nach unterstützten Anzeigenmarkierungen analysiert wird. Die folgenden Abschnitte beschreiben, welche Marker MediaTailor verwendet.

EXT-X-ANLAGE

Das EXT-X-ASSET-Tag enthält Metadaten, die vom Ad Decision Server (ADS) zum Personalisieren von Inhalten für die Anzeige verwendet werden. EXT-X-ASSET-Parameter sind durch Kommas getrennte Schlüssel-Wert-Paare.

Um diesen Tag verwenden zu können, müssen Sie die folgenden Anforderungen erfüllen:

- Sie müssen die EXT-X-ASSET-Werte im Ursprungsmanifest URL-kodieren. Es folgt das Beispiel eines EXT-X-ASSET-Tags mit Schlüsseln und URL-kodierten Werten.

```
#EXT-X-ASSET:GENRE=CV,CAID=12345678,EPIISODE="Episode%20Name%20Date",SEASON="Season%20Name%20and%20Number",SERIES="Series%2520Name"
```

- Sie müssen die dynamische [asset.] Variable und die Schlüssel in Ihre MediaTailor ADS-Konfiguration aufnehmen. Das folgende Beispiel zeigt eine MediaTailor ADS-Konfiguration mit der dynamischen [asset.] Variablen und den Schlüsseln.

```
https://myads.com/stub?  
c=[asset.GENRE]&g=[asset.CAID]&e=[asset.EPISODE]&s=[asset.SEASON]&k=[asset.SERIES]
```

Beispiel VAST-Anfrage

Es folgt das Beispiel einer VAST-GET-Anforderung an ein ADS.

```
https://myads.com/stub?c=CV&g=12345678&e=Episode%20Name%20Date&s=Season  
%20Name%20and%20Number&k=Series%2520Name
```

EXT-X-CUE-OUT und EXT-X-CUE-IN

Diese Art von Werbe-Marker ist die geläufigste. Die folgenden Beispiele zeigen Optionen für diese Cue-Marker.

```
#EXT-X-CUE-OUT:DURATION=120  
...  
#EXT-X-CUE-IN
```

```
#EXT-X-CUE-OUT:30.000  
...  
#EXT-X-CUE-IN
```

```
#EXT-X-CUE-OUT  
...  
#EXT-X-CUE-IN
```

EXT-X-DATERANGE

Bei EXT-X-DATERANGE Werbemarkierungs-Tags verwenden Sie SCTE35-OUT-Attribute, um den Zeitpunkt der Werbeverfügbarkeit festzulegen.

Note

AWS Elemental MediaTailor ignoriert alle START-DATE Attribute, die für EXT-X-DATERANGE Anzeigenmarkierungen bereitgestellt werden.

Sie können den Ad-Avail auf eine der folgenden Arten angeben:

- EXT-X-DATERANGE-Tag mit SCTE35-OUT- und DURATION-Spezifikationen.

Beispiel

```
#EXT-X-DATERANGE:ID="splice-6FFFFFFF0",START-DATE="2019-01T00:15:00Z\n",DURATION=60.000,SCTE35-OUT=0xF
```

- Gepaarte EXT-X-DATERANGE-Tags, das erste mit einer SCTE35-OUT-Spezifikation und das zweite mit einer SCTE35-IN-Spezifikation.

Beispiel

```
#EXT-X-DATERANGE:ID="splice-6FFFFFFF0",START-DATE="2019-01T00:15:00Z\n",SCTE35-OUT=0xF
...
#EXT-X-DATERANGE:ID="splice-6FFFFFFF0",START-DATE="2019-01T00:15:00Z\n",SCTE35-IN=0xF
```

- Eine Kombination aus den vorherigen Optionen. Sie geben ein EXT-X-DATERANGE-Tag mit SCTE35-OUT- und DURATION-Spezifikationen gefolgt von einem EXT-X-DATERANGE-Tag mit einer SCTE35-IN-Spezifikation an. MediaTailor verwendet in diesem Fall die früheste Cue-In-Einstellung aus den beiden Spezifikationen.

Beispiel

```
#EXT-X-DATERANGE:ID="splice-6FFFFFFF0",START-DATE="2019-01T00:15:00Z\n",DURATION=60.000,SCTE35-OUT=0xF
...
#EXT-X-DATERANGE:ID="splice-6FFFFFFF0",START-DATE="2019-01T00:15:00Z\n",SCTE35-IN=0xF
```


EXT-X-SPLICEPOINT- SCTE35

Sie hängen das EXT-X-SPLICEPOINT-SCTE35-Werbemarkier-Tag mit einer SCTE-35-Nutzlast im base64-kodierten binären Format an. Das dekodierte binäre Format muss einen SCTE-35-splice_info_section mit dem Cue-out-Marker 0x34 als Anfangspunkt der Platzierungsmöglichkeit für Anbieter und dem Cue-in-Marker 0x35 als Endpunkt der Platzierungsmöglichkeit für Anbieter enthalten.

Das folgende Beispiel zeigt die Splice-Punkt-Spezifikation mit base64-kodierten binären Nutzlasten, die die Cue-out- und Cue-in-Marker angeben.

```
#EXT-X-SPLICEPOINT-SCTE35:/DA9AAAAAAAAAAP/wBQb+uYbZqwAnAiVDVUVJAAAKqX//
AAEjW4AMEU1EU05CMDAxMTMyMjE5M190NAAAmXz5JA==
...
#EXT-X-SPLICEPOINT-SCTE35:/DA4AAAAAAAAAAP/wBQb+tTaaAwAiAiBDVUVJAAAKqH+/
DBFNRFNOQjAwMTEzMjIxOTJfTjUAAIiGK1s=
```

Passthrough für Werbemarkierungen aktivieren

Standardmäßig enthalten MediaTailor personalisierte Manifeste für HLS nicht die SCTE-35-Werbemarkierungen aus den ursprünglichen Manifesten. Wenn das Passthrough für Anzeigenmarkierungen aktiviert ist, werden die folgenden Werbemarkierungen von den ursprünglichen Manifesten in personalisierte Manifeste weitergeleitet:

- EXT-X-CUE-IN
- EXT-X-CUE-RAUS
- EXT-X-SPLICEPOINT- SCTE35

Der Passthrough für Werbemarkierungen ist eine optionale Einstellung. Verwenden Sie Passthrough für Werbemarkierungen, wenn Sie möchten, dass die SCTE-Werbemarkierungen in das MediaTailor personalisierte Manifest aufgenommen werden. Zu den häufigsten Anwendungsfällen gehören die folgenden:

- Austausch von Inhalten — Führen Sie eine Inhaltsersetzung oder eine Inhaltsbeschränkung durch.
- Anzeigenverfolgung — Geben Sie Informationen zur Anzeigenverfolgung auf Grundlage des Vorhandenseins oder Fehlens einer oder mehrerer Werbemarkierungen an.

- Player-Einstellungen — Aktiviere die Scrubbing- oder Countdown-Timer-Funktionen in der Benutzeroberfläche des Players, je nachdem, ob Werbemarkierungen vorhanden sind oder nicht.

Note

MediaTailor ändert die Werte für diese Markierungen nicht. Wenn EXT-X-CUE-OUT beispielsweise 60 im Ursprungsmanifest der Wert 0 steht, aber keine Werbung geschaltet MediaTailor wird, wird der Wert 0 im personalisierten Manifest nicht geändert.

Aktivieren Sie den Passthrough für Werbemarkierungen

Sie können den Passthrough für Werbemarkierungen mit dem AWS Management Console oder dem AWS Command Line Interface (AWS CLI) aktivieren.

So aktivieren Sie den Passthrough für Werbemarkierungen mithilfe der Konsole

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>
2. Wählen Sie entweder Neue Konfiguration oder Konfiguration bearbeiten aus.
3. Wählen Sie im Bereich Erweiterte Einstellungen im Dropdownmenü die Option Aktivieren aus.

Um den Passthrough für Werbemarkierungen mit dem AWS Command Line Interface (AWS CLI) zu aktivieren

Verwenden Sie den [put-playback-configuration](#)-Befehl.

Behandlung von HLS-Manifest-Tags

In diesem Abschnitt wird beschrieben, wie Tags im personalisierten Ausgabemanifest AWS Elemental MediaTailor verwaltet werden.

EXT-X-CUE Tags

MediaTailor ersetzt EXT-X-CUE-OUT, EXT-X-CUE-OUT-CONT, und EXT-X-CUE-IN -Tags im Eingabemanifest durch EXT-X-DISCONTINUITY Tags im Ausgabemanifest. Die DISCONTINUITY-Tags markieren die folgenden Grenzen:

- Wo der Hauptinhalt zu einer Werbung wechselt

- Übergang von einer Werbung zur anderen
- Wo eine Werbung zum Hauptinhalt zurückkehrt

EXT-X-DATERANGE tags

MediaTailor leitet EXT-X-DATERANGE Tags vom Eingabemanifest zum Ausgabemanifest weiter. MediaTailor fügt außerdem EXT-X-DISCONTINUITY Tags ein, die den DATERANGE Tags entsprechen. Die DISCONTINUITY-Tags markieren die folgenden Grenzen:

- Wo der Hauptinhalt zu einer Werbung wechselt
- Übergang von einer Werbung zur anderen
- Wo eine Werbung zum Hauptinhalt zurückkehrt

EXT-X-KEY tags

MediaTailor durchläuft EXT-X-KEY Tags aus dem Eingabemanifest. Diese Tags zeigen an, dass der Hauptinhalt verschlüsselt ist. Da Werbung nicht verschlüsselt sind, fügt MediaTailor EXT-X-KEY:METHOD=NONE zu Beginn einer Werbung ein. Wenn die Wiedergabe zum Hauptinhalt zurückkehrt, wird die Verschlüsselung MediaTailor wieder aktiviert, indem das EXT-X-KEY Tag mit dem als METHOD Verschlüsselungstyp definierten Wert eingefügt wird.

Unbekannte Tags

MediaTailor durchläuft alle unbekannten und benutzerdefinierten Tags vom Eingabemanifest zum Ausgabemanifest.

Beispiele für HLS-Manifeste

Die folgenden Abschnitte enthalten Beispiele für HLS-Manifeste und personalisierte Manifeste.

Beispiele für HLS-Ursprungsmanifeste

Das folgende Beispiel zeigt ein HLS-Mastermanifest, das HLS vom Inhaltsursprung AWS Elemental MediaTailor empfangen hat.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-INDEPENDENT-SEGMENTS
```

```
#EXT-X-STREAM-INF:BANDWIDTH=2665726,AVERAGE-
BANDWIDTH=2526299,RESOLUTION=960x540,FRAME-
RATE=29.970,CODECS="avc1.640029,mp4a.40.2",SUBTITLES="subtitles"
index_1.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=3956044,AVERAGE-
BANDWIDTH=3736264,RESOLUTION=1280x720,FRAME-
RATE=29.970,CODECS="avc1.640029,mp4a.40.2",SUBTITLES="subtitles"
index_2.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=995315,AVERAGE-
BANDWIDTH=951107,RESOLUTION=640x360,FRAME-
RATE=29.970,CODECS="avc1.4D401E,mp4a.40.2",SUBTITLES="subtitles"
index_3.m3u8
#EXT-X-MEDIA:TYPE=SUBTITLES,GROUP-
ID="subtitles",NAME="caption_1",DEFAULT=YES,AUTOSELECT=YES,FORCED=NO,LANGUAGE="eng",URI="index_
```

Das folgende Beispiel zeigt ein HLS-Medienmanifest, das von HLS vom AWS Elemental MediaTailor Inhaltsursprung empfangen wurde. In diesem Beispiel werden die Tags EXT-X-CUE-OUT und EXT-X-CUE-IN verwendet, um Ad-Avail-Gelegenheiten zu beschreiben.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:8779957
#EXTINF:6.006,
index_1_8779957.ts?m=1566416212
#EXTINF:6.006,
index_1_8779958.ts?m=1566416212
#EXTINF:5.372,
index_1_8779959.ts?m=1566416212
#EXT-OATCLS-SCTE35:/DA1AAAAAsvhAP/wFAXwAAAGf+/+AdLfiP4AG3dAAAEBQAAXytxmQ==
#EXT-X-CUE-OUT:20.020
#EXTINF:0.634,
index_1_8779960.ts?m=1566416212
#EXT-X-CUE-OUT-CONT:ElapsedTime=0.634,Duration=21,SCTE35=/DA1AAAAAsvhAP/wFAXwAAAGf
+/+AdLfiP4AG3dAAAEBQAAXytxmQ==
#EXTINF:6.006,
index_1_8779961.ts?m=1566416212
#EXT-X-CUE-OUT-CONT:ElapsedTime=6.640,Duration=21,SCTE35=/DA1AAAAAsvhAP/wFAXwAAAGf
+/+AdLfiP4AG3dAAAEBQAAXytxmQ==
#EXTINF:6.006,
index_1_8779962.ts?m=1566416212
#EXT-X-CUE-OUT-CONT:ElapsedTime=12.646,Duration=21,SCTE35=/DA1AAAAAsvhAP/wFAXwAAAGf
+/+AdLfiP4AG3dAAAEBQAAXytxmQ==
```

```
#EXTINF:6.006,
index_1_8779963.ts?m=1566416212
#EXT-X-CUE-OUT-CONT:ElapsedTime=18.652,Duration=21,SCTE35=/DA1AAAAAsvhAP/wFAXwAAAGf
+ /+AdLfIP4AG3dAAAEBQAAXytxmQ==
#EXTINF:1.368,
index_1_8779964.ts?m=1566416212
#EXT-X-CUE-IN
#EXTINF:4.638,
index_1_8779965.ts?m=1566416212
#EXTINF:6.006,
index_1_8779966.ts?m=1566416212
#EXTINF:6.006,
index_1_8779967.ts?m=1566416212
#EXTINF:6.006,
index_1_8779968.ts?m=1566416212
```

Beispiele für personalisierte HLS-Manifeste

Das folgende Beispiel zeigt ein HLS-Mastermanifest, das von AWS Elemental MediaTailor personalisiert wurde.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-MEDIA:LANGUAGE="eng",AUTOSELECT=YES,FORCED=NO,TYPE=SUBTITLES,URI=" ../../../../
manifest/43f3e412052f2808dd84ea1da90e92e914edddde/external-
canary-hls/ee1696a8-4f7f-4c4c-99de-9821131847e8/3.m3u8",GROUP-
ID="subtitles",DEFAULT=YES,NAME="caption_1"
#EXT-X-INDEPENDENT-SEGMENTS
#EXT-X-STREAM-INF:CODECS="avc1.640029,mp4a.40.2",AVERAGE-
BANDWIDTH=2526299,RESOLUTION=960x540,SUBTITLES="subtitles",FRAME-
RATE=29.97,BANDWIDTH=2665726
 ../../../../manifest/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0.m3u8
#EXT-X-STREAM-INF:CODECS="avc1.640029,mp4a.40.2",AVERAGE-
BANDWIDTH=3736264,RESOLUTION=1280x720,SUBTITLES="subtitles",FRAME-
RATE=29.97,BANDWIDTH=3956044
 ../../../../manifest/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/1.m3u8
#EXT-X-STREAM-INF:CODECS="avc1.4D401E,mp4a.40.2",AVERAGE-
BANDWIDTH=951107,RESOLUTION=640x360,SUBTITLES="subtitles",FRAME-
RATE=29.97,BANDWIDTH=995315
 ../../../../manifest/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/2.m3u8
```

Das folgende Beispiel zeigt ein Medienmastermanifest, das von AWS Elemental MediaTailor personalisiert wurde.

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:8779957
#EXT-X-DISCONTINUITY-SEQUENCE:0
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779957.ts?m=1566416212
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779958.ts?m=1566416212
#EXTINF:5.372,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779959.ts?m=1566416212
#EXT-X-DISCONTINUITY
#EXTINF:3.066667,
../.../.../segment/43f3e412052f2808dd84ea1da90e92e914eddddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779960
#EXTINF:3.0,
../.../.../segment/43f3e412052f2808dd84ea1da90e92e914eddddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779961
#EXTINF:3.0,
../.../.../segment/43f3e412052f2808dd84ea1da90e92e914eddddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779962
#EXTINF:3.0,
../.../.../segment/43f3e412052f2808dd84ea1da90e92e914eddddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779963
#EXTINF:2.966667,
../.../.../segment/43f3e412052f2808dd84ea1da90e92e914eddddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779964
#EXT-X-DISCONTINUITY
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779963.ts?m=1566416212
#EXTINF:1.368,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779964.ts?m=1566416212
#EXTINF:4.638,
```

```

https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779965.ts?m=1566416212
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779966.ts?m=1566416212
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779967.ts?m=1566416212
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779968.ts?m=1566416212

```

Integration einer MPEG-DASH-Quelle

AWS Elemental MediaTailor unterstützt .mpd Live- und Video-on-Demand (VOD) -Manifeste, die den Richtlinien für das dynamische DASH-Profil entsprechen. MediaTailor akzeptiert DASH-konforme Manifesteingaben für mehrere Perioden und einzelne Perioden und liefert DASH-konforme Manifestausgaben für mehrere Perioden.

Eingabemanifeste müssen Folgendes aufweisen:

- SCTE-35 Event-Streams mit Splice-Info-Einstellungen für `splice insert` oder `time signal`. Die Einstellungen können im transparenten XML- oder im base64-kodierten, binären Format angegeben werden.
- `Segment templates` mit `segment timelines`.

MediaTailor erfordert für veröffentlichte Manifeste, dass bei Aktualisierungen vom Ursprungsserver Folgendes unverändert bleibt:

- Die im Attribut `start` angegebenen Startzeiten von Zeiträumen.
- Werte von `presentationTimeOffset` in den Segmentvorlagen der Zeitraumdarstellungen.

Es hat sich bewährt, für die Werbeanzeige dieselben `AdaptationSet Representation` Einstellungen wie für die Zeiträume des Content-Streams festzulegen. AWS Elemental MediaTailor verwendet diese Einstellungen, um die Anzeigen so zu transkodieren, dass sie dem Inhaltsstream entsprechen, sodass ein reibungsloser Wechsel zwischen beiden möglich ist.

In den folgenden Abschnitten finden Sie weitere Informationen zum MediaTailor Umgang mit Anzeigen in DASH-Manifesten.

Themen

- [DASH-Werbemarkierungen](#)
- [DASH und Verfügbarkeitsdauer](#)
- [Segmentnummerierung des DASH-Manifests](#)
- [Beispiele für Live-DASH-Manifeste](#)
- [Beispiele für VOD-DASH-Manifeste](#)
- [DASH-Ortungsfunktion](#)

DASH-Werbemarkierungen

MediaTailor

AWS Elemental MediaTailor verwendet SCTE-35-Cue-Out-Markierungen, um verfügbare Werbeanzeigen im DASH-Manifest anhand der folgenden Logik zu identifizieren:

- DASH mit mehreren Perioden: MediaTailor fügt Anzeigen für die jeweils erste Anzeige ein, die entweder oder Event Cue-Out-Markierungen Period enthält. `SpliceInsert TimeSignal` MediaTailor ignoriert zusätzliche Event Markierungen in der. Period
- Einzeiliger DASH: MediaTailor fügt jeweils Anzeigen Event in die einPeriod, die entweder `SpliceInsert` oder `TimeSignal` Cue-Out-Markierungen enthalten.

AWS Elemental MediaTailor verwaltet DASH-Manifeste standardmäßig als Manifeste mit mehreren Perioden. Sie können die Konfiguration so ändern, dass sie DASH-Manifeste mit einem einzigen Zeitraum von Ihrem Ursprungs-Server verarbeitet. Weitere Informationen finden Sie unter [the section called “Konfiguration erstellen”](#).

Die folgenden Abschnitte enthalten zusätzliche Informationen zur Handhabung von DASH-Werbemarkern und enthalten dekorierte Manifeste von Anfang an.

XML-Anforderungen für das DASH-Ursprungsmanifest

Anzeigenmarkierungen in DASH-Manifesten aus dem Ursprungsland müssen korrekt formatiert sein, MediaTailor damit Werbeunterbrechungen identifiziert werden können. In den folgenden Themen werden diese Formatierungsanforderungen in klarem XML beschrieben.

SpliceInsert in klarem XML

`SpliceInsert` Werbemarkierungen in klarem XML müssen Folgendes enthalten:

- EventStream muss das folgende Attribut haben: `schemeIdUri=urn:scte:scte35:2013:xml`
- Event muss halten `scte35:SpliceInfoSection`
- `scte35:SpliceInfoSection` muss halten `scte35:SpliceInsert`
- `scte35:SpliceInsert` muss das folgende Attribut haben: `outOfNetworkIndicator="true"`

Example **SpliceInsert** in XML

Im folgenden Beispiel sind die erforderlichen SCTE-Markierungen fett gedruckt.

```
<Period start="PT444806.040S" id="123586" duration="PT15.000S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="1350000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="180832" tier="4095">
        <scte35:SpliceInsert spliceEventId="4026531855"
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
          <scte35:Program><scte35:SpliceTime ptsTime="5672624400"/></scte35:Program>
          <scte35:BreakDuration autoReturn="true" duration="1350000"/>
        </scte35:SpliceInsert>
      </scte35:SpliceInfoSection>
    </Event>
  .
  .
  .
</Period>
```

TimeSignal in klarem XML

TimeSignal Werbemarkierungen in klarem XML müssen Folgendes enthalten:

- EventStream muss das folgende Attribut haben: `schemeIdUri=urn:scte:scte35:2013:xml`
- Event muss halten `scte35:SpliceInfoSection`
- `scte35:SpliceInfoSection` muss halten `scte35:TimeSignal`
- `scte35:SpliceInfoSection` muss auch halten `scte35:SegmentationDescriptor`
- `scte35:SegmentationDescriptor` muss das folgende Attribut haben, wobei der Wert gültig ist [Cue-Out-Zahlen](#): `segmentationTypeId="xx"`
- `scte35:SegmentationDescriptor` muss gelten `scte35:SegmentationUpid`

Cue-Out-Zahlen

Die folgenden Cue-Out-Nummern werden für die unterstützt. `segmentationTypeId`

Segmentierungsnachricht	segmentationTypeId Wert	Hexadezimaler Wert
Ende der Werbung für den Vertriebspartner	51	0x51
Beginn der Werbung für Vertriebspartner	50	0x32
Ende der Platzierungsmöglichkeit für Vertriebspartner	55	0x37
Beginn der Platzierungsmöglichkeit für Vertriebspartner	54	0x36
Pause beenden	35	0x23
Ende der Anbieter-Werbung	49	0 x 31
Beginn der Anbieterwerbung	48	0x30
Ende der Platzierungsmöglichkeit im Provider-Overlay	57	0x39
Beginn der Platzierungsmöglichkeit im Provider-Overlay	56	0x38
Ende der Platzierungsmöglichkeit für Anbieter	53	0x35
Beginn der Platzierungsmöglichkeit für Anbieter	52	0x34
Pause starten	34	0 x 22

Example **TimeSignal** in XML

Im folgenden Beispiel sind die erforderlichen SCTE-Markierungen fett gedruckt.

```
<Period start="PT346530.250S" id="178443" duration="PT61.561S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="5310000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="183003" tier="4095">
        <scte35:TimeSignal>
          <scte35:SpliceTime ptsTime="3442857000"/>
        </scte35:TimeSignal>
        <scte35:SegmentationDescriptor segmentationEventId="1414668"
segmentationEventCancelIndicator="false"
segmentationDuration="8100000" segmentationTypeId="52" segmentNum="0"
segmentsExpected="0">
          <scte35:DeliveryRestrictions webDeliveryAllowedFlag="false"
noRegionalBlackoutFlag="false" archiveAllowedFlag="false" deviceRestrictions="3"/>
          <scte35:SegmentationUpid segmentationUpidType="12"
segmentationUpidLength="2">0100</scte35:SegmentationUpid>
        </scte35:SegmentationDescriptor>
      </scte35:SpliceInfoSection>
    </Event>
    .
    .
    .
  </EventStream>
</Period>
```

Base64-kodierte Binäranforderungen für das DASH-Ursprungsmanifest

Anzeigenmarkierungen in DASH-Manifesten aus dem Ursprungsland müssen korrekt formatiert sein, MediaTailor damit Werbeunterbrechungen identifiziert werden können. In den folgenden Themen werden diese Formatierungsanforderungen in einer Base64-codierten Binärdatei beschrieben.

TimeSignal Sowohl als auch **SpliceInsert** Werbemarkierungen in Base64-codierten Manifesten müssen Folgendes enthalten:

- EventStream muss das folgende Attribut haben: `urn:scte:scte35:2014:xml+bin`
- Event muss halten `scte35:Signal`
- `scte35:Signal` muss gelten `scte35:Binary`, das eine Base64-kodierte Binärdatei enthält.

Die dekodierte Binärdatei muss dieselben Informationen enthalten wie die, die für klare XML-Werbemarkierungen erforderlich sind. `splice_info_section`

- Der Befehlstyp muss entweder `splice_insert()` sein oder `time_signal()`
- Die zusätzlichen Einstellungen müssen den unter [TimeSignal in klarem XML](#) und beschriebenen Einstellungen entsprechen [SpliceInsert in klarem XML](#).

Das dekodierte binäre Format muss einen `splice_info_section` mit dem gleichen Satz von Informationen bereitstellen, der im transparenten XML-Format in einem `scte35:SpliceInfoSection`-Element ausgegeben würde. Der Befehlstyp muss entweder `splice_insert()` oder `time_signal()` sein, und die zusätzlichen Einstellungen müssen denen entsprechen, die zuvor für eine Bereitstellung im transparenten XML-Format beschrieben wurden.

Das folgende Beispiel zeigt diese Option, wobei die erforderlichen Marker fettgedruckt sind.

```
<Period start="PT444806.040S" id="123586" duration="PT15.000S">
  <EventStream schemeIdUri="urn:scte:scte35:2014:xml+bin" timescale="1">
    <Event presentationTime="1541436240" duration="24" id="29">
      <scte35:Signal xmlns="http://www.scte.org/schemas/35/2016">
        <scte35:Binary>DAhAAAAAAAAAAP/wEAUAAAHaf+9/fgAg9YDAAAAAAAAA25aoh</
scte35:Binary>
      </scte35:Signal>
    </Event>
    <Event presentationTime="1541436360" duration="24" id="30">
      <scte35:Signal xmlns="http://www.scte.org/schemas/35/2016">

        <scte35:Binary>QW5vdGhlciB0ZXN0IHN0cmVuZyBmb3IgaW5jb2RpbmcgdG8gQmFzZTY0IGVuY29kZWQgYm1uYXJ5J5Lg=
scte35:Binary>
        </scte35:Signal>
      </Event>
      .
      .
      .
    </Period>
```

Nachstehend wird das dekodierte binäre Format für das erste, im voranstehenden Beispiel aufgeführte Ereignis angegeben. Die Einstellung für `splice_command_type` lautet 5, wodurch `splice_insert` angegeben wird.

```
{
```

```

"table_id": 252,
"section_syntax_indicator": false,
"private_indicator": false,
"section_length": 33,
"protocol_version": 0,
"encrypted_packet": false,
"encryption_algorithm": 0,
"pts_adjustment": 0,
"cw_index": 0,
"tier": "0xFFF",
"splice_command_length": 16,
"splice_command_type": 5,
"splice_command": {
  "splice_event_id": 448,
  "splice_event_cancel_indicator": false,
  "out_of_network_indicator": true,
  "program_splice_flag": true,
  "duration_flag": true,
  "splice_immediate_flag": false,
  "utc_splice_time": {
    "time_specified_flag": false,
    "pts_time": null
  },
  "component_count": 0,
  "components": null,
  "break_duration": {
    "auto_return": false,
    "duration": {
      "pts_time": 2160000,
      "wall_clock_seconds": 24.0,
      "wall_clock_time": "00:00:24:00000"
    }
  },
  "unique_program_id": 49152,
  "avail_num": 0,
  "avails_expected": 0
},
"splice_descriptor_loop_length": 0,
"splice_descriptors": null,
"Scte35Exception": {
  "parse_status": "SCTE-35 cue parsing completed with 0 errors.",
  "error_messages": [],
  "table_id": 252,
  "splice_command_type": 5
}

```

```
}
}
```

DASH und Verfügbarkeitsdauer

Wenn bei der Wiedergabe AWS Elemental MediaTailor auf eine Anzeige gestoßen wird, wird die Anzeige ganz oder teilweise durch Werbung ersetzt. MediaTailor Startet das Ersetzen von Anzeigen zu Beginn der Verfügbarkeit der Anzeige und umfasst Anzeigen wie folgt:

- Wenn für die Anzeigenverfügbarkeit eine Dauer angegeben ist, werden so viele Anzeigen MediaTailor eingeschlossen, wie innerhalb der Zeitbegrenzung Platz finden, ohne dass der nachfolgende Inhalt überschrieben wird.
- Wenn keine Dauer angegeben ist, werden Anzeigen MediaTailor eingeschlossen, bis das Ende der verfügbaren Anzeige erreicht ist. Für Manifeste mit mehreren Zeiträumen ist dies das Ende des Zeitraums. Bei Manifesten mit einer Periode ist dies das Ende der Veranstaltung. MediaTailor spielt keine Werbung nach dem Ende der verfügbaren Anzeige ab und kürzt, wenn das Ende erreicht ist, die aktuelle Anzeige, anstatt den darauffolgenden Inhalt zu überschreiben.

Wie AWS Elemental MediaTailor sieht die Dauer der Anzeigenverfügbarkeit aus

AWS Elemental MediaTailor sucht in der folgenden Reihenfolge nach einer Einstellung für die Dauer:

1. `Event duration`
2. Für Splice-Insert-Marker unter `scte35:BreakDuration duration`
3. Für Zeitsignal-Marker unter `scte35:SegmentationDescriptor segmentationDuration`

Wenn AWS Elemental MediaTailor keine dieser Einstellungen gefunden wird, wird die Anzeigeninklusion ohne Dauer verwaltet.

Das folgende Beispiel zeigt ein Event mit einer `duration`.

```
<Period start="PT444806.040S" id="123586" duration="PT15.000S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="1350000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="180832"
tier="4095">
        <scte35:SpliceInsert spliceEventId="4026531855"
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
```

```

        <scte35:Program><scte35:SpliceTime ptsTime="5672624400"/></scte35:Program>
        <scte35:BreakDuration autoReturn="true" duration="1350000"/>
      </scte35:SpliceInsert>
    </scte35:SpliceInfoSection>
  </Event>
  ...

```

Das folgende Beispiel zeigt ein Ad-Avail ohne angegebene Dauer. Das Event besitzt keine duration und das Element scte35:SpliceInsert enthält kein untergeordnetes scte35:BreakDuration-Element.

```

<Period start="PT444836.720S" id="123597" duration="PT12.280S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event>
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="180832"
tier="4095">
        <scte35:SpliceInsert spliceEventId="4026531856"
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
          <scte35:Program><scte35:SpliceTime ptsTime="5675385600"/></scte35:Program>
        </scte35:SpliceInsert>
      </scte35:SpliceInfoSection>
    </Event>
    ...
  </EventStream>
</Period>

```

Segmentnummerierung des DASH-Manifests

MediaTailor unterstützt Mediensegmente<SegmentTemplate>, die mit <SegmentTimeline> und dem media Attribut definiert sind. Sie können die Liste der Mediensegmente entweder mit dem Bezeichner \$Number\$ oder dem Bezeichner \$Time\$ im Attribut media angeben.

Das folgende Beispiel zeigt eine SegmentTemplate mit einer media-Attributeinstellung, die den Bezeichner \$Number\$ verwendet.

```

<SegmentTemplate initialization="index_subtitles_4_0_init.mp4?
m=1532451703" media="index_subtitles_4_0_$Number$.mp4?m=1532451703"
presentationTimeOffset="1062336677920" startNumber="2349899" timescale="90000">
  <SegmentTimeline>
    <S d="540540" r="2" t="1062338840080"/>
    <S d="69069" t="1062340461700"/>
  </SegmentTimeline>
</SegmentTemplate>

```

```

    </SegmentTimeline>
  </SegmentTemplate>

```

Das folgende Beispiel zeigt eine SegmentTemplate mit einer media-Attributeinstellung, die den Bezeichner \$Time\$ verwendet.

```

<SegmentTemplate
  initialization="asset_720p_8000K_9_init.mp4" media="asset_720p_8000K_9_$Time$.mp4"
  startNumber="1" timescale="90000">
  <SegmentTimeline>
    <S d="180000" r="2" t="0"/>
    <S d="147000" t="540000"/>
  </SegmentTimeline>
</SegmentTemplate>

```

Beispiele für Live-DASH-Manifeste

Dieser Abschnitt enthält Beispiele für Live-DASH-Manifeste. In jedem Beispiel wird ein Manifest aufgeführt, das vom Ursprungsserver empfangen wurde. Danach MediaTailor wurde das Manifest mit Werbung personalisiert.

Themen

- [Beispiel für das Einfügen eines DASH-Manifest-Splice](#)
- [Beispiel für ein Zeitsignal im DASH-Manifest](#)
- [Base64-kodiertes Binärbeispiel für das DASH-Manifest mit einperiodischer Eingabe](#)

Beispiel für das Einfügen eines DASH-Manifest-Splice

Beispiel des DASH-Ursprungs-Manifests für Splice-Insert-Marker

Das folgende Beispiel aus einem MPD-Manifest zeigt den Ad Avail-Zeitraum in einem Manifest, das von DASH vom Ursprungs-Server der Inhalte empfangen wurde. T

```

<Period start="PT173402.036S" id="46041">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="9450000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="183265"
tier="4095">

```



```

        <scte35:SpliceInsert spliceEventId="99"
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
        <scte35:Program><scte35:SpliceTime ptsTime="7835775000"/></
scte35:Program>
        <scte35:BreakDuration autoReturn="true" duration="9450000"/>
        </scte35:SpliceInsert>
        </scte35:SpliceInfoSection>
    </Event>
</EventStream>
    <AdaptationSet mimeType="video/mp4" segmentAlignment="true"
subsegmentAlignment="true" startWithSAP="1" subsegmentStartsWithSAP="1"
bitstreamSwitching="true">
        <Representation id="1" width="640" height="360" frameRate="30/1"
bandwidth="749952" codecs="avc1.4D4029">
            <SegmentTemplate timescale="30" media="index_video_1_0_$Number
$.mp4?m=1531257079" initialization="index_video_1_0_init.mp4?m=1531257079"
startNumber="46042" presentationTimeOffset="5202061">
                <SegmentTimeline>
                    <S t="5202061" d="115"/>
                    <S t="5202176" d="120" r="4"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation id="2" width="1280" height="720" frameRate="30/1"
bandwidth="2499968" codecs="avc1.4D4029">
            <SegmentTemplate timescale="30" media="index_video_3_0_$Number
$.mp4?m=1531257079" initialization="index_video_3_0_init.mp4?m=1531257079"
startNumber="46042" presentationTimeOffset="5202061">
                <SegmentTimeline>
                    <S t="5202061" d="115"/>
                    <S t="5202176" d="120" r="4"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation id="3" width="1920" height="1080" frameRate="30/1"
bandwidth="4499968" codecs="avc1.4D4029">
            <SegmentTemplate timescale="30" media="index_video_5_0_$Number
$.mp4?m=1531257079" initialization="index_video_5_0_init.mp4?m=1531257079"
startNumber="46042" presentationTimeOffset="5202061">
                <SegmentTimeline>
                    <S t="5202061" d="115"/>
                    <S t="5202176" d="120" r="4"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>
</MPD>

```

```

        </SegmentTemplate>
    </Representation>
</AdaptationSet>
<AdaptationSet mimeType="audio/mp4" segmentAlignment="0" lang="eng">
    <Representation id="4" bandwidth="128858" audioSamplingRate="44100"
codecs="mp4a.40.2">
        <SegmentTemplate timescale="44100" media="index_audio_2_0_${Number
$.mp4?m=1531257079" initialization="index_audio_2_0_init.mp4?m=1531257079"
startNumber="46042" presentationTimeOffset="7647030507">
            <SegmentTimeline>
                <S t="7647030507" d="168959"/>
                <S t="7647199468" d="176127" r="1"/>
                <S t="7647551723" d="177151"/>
                <S t="7647728875" d="176127" r="1"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation id="5" bandwidth="128858" audioSamplingRate="44100"
codecs="mp4a.40.2">
        <SegmentTemplate timescale="44100" media="index_audio_4_0_${Number
$.mp4?m=1531257079" initialization="index_audio_4_0_init.mp4?m=1531257079"
startNumber="46042" presentationTimeOffset="7647030507">
            <SegmentTimeline>
                <S t="7647030507" d="168959"/>
                <S t="7647199468" d="176127" r="1"/>
                <S t="7647551723" d="177151"/>
                <S t="7647728875" d="176127" r="1"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation id="6" bandwidth="128858" audioSamplingRate="44100"
codecs="mp4a.40.2">
        <SegmentTemplate timescale="44100" media="index_audio_6_0_${Number
$.mp4?m=1531257079" initialization="index_audio_6_0_init.mp4?m=1531257079"
startNumber="46042" presentationTimeOffset="7647030507">
            <SegmentTimeline>
                <S t="7647030507" d="168959"/>
                <S t="7647199468" d="176127" r="1"/>
                <S t="7647551723" d="177151"/>
                <S t="7647728875" d="176127" r="1"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>

```

```
</Period>
```

Beispiel einer personalisierten DASH-Antwort für Splice-Insert-Marker

AWS Elemental MediaTailor personalisiert die Werbeanzeige mit Werbespezifikationen. Die Personalisierungen spiegeln die vom Player empfangenen Daten zum Betrachter wider sowie die Werbekampagnen, die derzeit ausgeführt werden.

Das folgende Beispiel zeigt eine Anzeige, die nach MediaTailor der Personalisierung verfügbar ist.

```
<Period id="46041_1" start="PT48H10M2.036S">
  <BaseURL>http://cdnlocation.net/EXAMPLE_PRODUCT/</BaseURL>
  <AdaptationSet bitstreamSwitching="false" frameRate="30/1" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
    <SegmentTemplate startNumber="1" timescale="90000"/>
    <Representation bandwidth="10000000" codecs="avc1.640028" height="1080"
id="1" width="1920">
      <SegmentTemplate initialization="EXAMPLE_PRODUCT_1080p_10init.mp4"
media="EXAMPLE_PRODUCT_1080p_10_${Number%09d$.mp4" startNumber="1"
timescale="90000"><SegmentTimeline><S d="180000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate>
    </Representation>
    <Representation bandwidth="4000000" codecs="avc1.64001f" height="720"
id="2" width="1280">
      <SegmentTemplate initialization="EXAMPLE_PRODUCT_720p_9init.mp4"
media="EXAMPLE_PRODUCT_720p_9_${Number%09d$.mp4" startNumber="1"
timescale="90000"><SegmentTimeline><S d="180000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate>
    </Representation>
    <Representation bandwidth="2500000" codecs="avc1.64001f" height="720"
id="3" width="1280">
      <SegmentTemplate initialization="EXAMPLE_PRODUCT_720p_8init.mp4"
media="EXAMPLE_PRODUCT_720p_8_${Number%09d$.mp4" startNumber="1"
timescale="90000"><SegmentTimeline><S d="180000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate>
    </Representation>
    <Representation bandwidth="2000000" codecs="avc1.64001f" height="540"
id="4" width="960">
      <SegmentTemplate initialization="EXAMPLE_PRODUCT_540p_7init.mp4"
media="EXAMPLE_PRODUCT_540p_7_${Number%09d$.mp4" startNumber="1"
```

```

    timescale="90000"><SegmentTimeline><S d="180000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate>
    </Representation>
    <Representation bandwidth="1350000" codecs="avc1.64001e" height="396"
id="5" width="704">
        <SegmentTemplate initialization="EXAMPLE_PRODUCT_396p_6init.mp4"
media="EXAMPLE_PRODUCT_396p_6_${Number%09d$.mp4" startNumber="1"
timescale="90000"><SegmentTimeline><S d="180000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate>
    </Representation>
    <Representation bandwidth="900000" codecs="avc1.64001e" height="396" id="6"
width="704">
        <SegmentTemplate initialization="EXAMPLE_PRODUCT_396p_5init.mp4"
media="EXAMPLE_PRODUCT_396p_5_${Number%09d$.mp4" startNumber="1"
timescale="90000"><SegmentTimeline><S d="180000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate>
    </Representation>
    <Representation bandwidth="600000" codecs="avc1.64001e" height="396" id="7"
width="704">
        <SegmentTemplate initialization="EXAMPLE_PRODUCT_396p_4init.mp4"
media="EXAMPLE_PRODUCT_396p_4_${Number%09d$.mp4" startNumber="1"
timescale="90000"><SegmentTimeline><S d="180000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate>
    </Representation>
    <Representation bandwidth="450000" codecs="avc1.640016" height="288" id="8"
width="512">
        <SegmentTemplate initialization="EXAMPLE_PRODUCT_288p_3init.mp4"
media="EXAMPLE_PRODUCT_288p_3_${Number%09d$.mp4" startNumber="1"
timescale="90000"><SegmentTimeline><S d="180000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate>
    </Representation>
    <Representation bandwidth="300000" codecs="avc1.640016" height="288" id="9"
width="512">
        <SegmentTemplate initialization="EXAMPLE_PRODUCT_288p_2init.mp4"
media="EXAMPLE_PRODUCT_288p_2_${Number%09d$.mp4" startNumber="1"
timescale="90000"><SegmentTimeline><S d="180000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate>
    </Representation>
    <Representation bandwidth="200000" codecs="avc1.640016" height="288"
id="10" width="512">
        <SegmentTemplate initialization="EXAMPLE_PRODUCT_288p_1init.mp4"
media="EXAMPLE_PRODUCT_288p_1_${Number%09d$.mp4" startNumber="1"
timescale="90000"><SegmentTimeline><S d="180000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate>

```

```

        </Representation>
    </AdaptationSet>
    <AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
        <SegmentTemplate initialization="EXAMPLE_PRODUCT_audio_aac_a1_128kinit.mp4"
media="EXAMPLE_PRODUCT_audio_aac_a1_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"/>
        <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="11"><SegmentTemplate
initialization="EXAMPLE_PRODUCT_audio_aac_a1_128kinit.mp4"
media="EXAMPLE_PRODUCT_audio_aac_a1_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"><SegmentTimeline><S d="96000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate></Representation>
    </AdaptationSet>
    <AdaptationSet lang="enm" mimeType="audio/mp4" segmentAlignment="0">
        <SegmentTemplate initialization="EXAMPLE_PRODUCT_audio_aac_a2_128kinit.mp4"
media="EXAMPLE_PRODUCT_audio_aac_a2_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"/>
        <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="12"><SegmentTemplate
initialization="EXAMPLE_PRODUCT_audio_aac_a2_128kinit.mp4"
media="EXAMPLE_PRODUCT_audio_aac_a2_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"><SegmentTimeline><S d="96000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate></Representation>
    </AdaptationSet>
    <AdaptationSet lang="por" mimeType="audio/mp4" segmentAlignment="0">
        <SegmentTemplate initialization="EXAMPLE_PRODUCT_audio_aac_a3_128kinit.mp4"
media="EXAMPLE_PRODUCT_audio_aac_a3_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"/>
        <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="13"><SegmentTemplate
initialization="EXAMPLE_PRODUCT_audio_aac_a3_128kinit.mp4"
media="EXAMPLE_PRODUCT_audio_aac_a3_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"><SegmentTimeline><S d="96000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate></Representation>
    </AdaptationSet>
    <AdaptationSet lang="spa" mimeType="audio/mp4" segmentAlignment="0">
        <SegmentTemplate initialization="EXAMPLE_PRODUCT_audio_aac_a4_128kinit.mp4"
media="EXAMPLE_PRODUCT_audio_aac_a4_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"/>
        <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="14"><SegmentTemplate
initialization="EXAMPLE_PRODUCT_audio_aac_a4_128kinit.mp4"
media="EXAMPLE_PRODUCT_audio_aac_a4_128k_${Number%09d$.mp4" startNumber="1"

```

```

    timescale="48000"><SegmentTimeline><S d="96000" r="11" t="0"/></SegmentTimeline></
SegmentTemplate></Representation>
    </AdaptationSet>
  </Period>

```

Beispiel für ein Zeitsignal im DASH-Manifest

Beispiel des DASH-Ursprungs-Manifests für Zeitsignal-Marker

Das folgende Beispiel zeigt den Ad Avail-Zeitraum in einem Manifest, das von DASH vom Ursprungs-Server der Inhalte empfangen wurde. Das folgende Beispiel zeigt die `scte35:TimeSignal`-Marker.

```

<Period start="PT346530.250S" id="178443" duration="PT61.561S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="5310000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="183003"
tier="4095">
        <scte35:TimeSignal>
          <scte35:SpliceTime ptsTime="3442857000"/>
        </scte35:TimeSignal>
        <scte35:SegmentationDescriptor segmentationEventId="1414668"
segmentationEventCancelIndicator="false" segmentationDuration="8100000">
          <scte35:DeliveryRestrictions webDeliveryAllowedFlag="false"
noRegionalBlackoutFlag="false" archiveAllowedFlag="false" deviceRestrictions="3"/>
          <scte35:SegmentationUpid segmentationUpidType="12"
segmentationUpidLength="2" segmentationTypeId="52" segmentNum="0"
segmentsExpected="0">0100</scte35:SegmentationUpid>
        </scte35:SegmentationDescriptor>
      </scte35:SpliceInfoSection>
    </Event>
  </EventStream>
  <AdaptationSet mimeType="video/mp4" segmentAlignment="true"
subsegmentAlignment="true" startWithSAP="1" subsegmentStartsWithSAP="1"
bitstreamSwitching="true">
    <Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="1000000" codecs="avc1.4D401F">
      <SegmentTemplate timescale="30000" media="index_video_1_0_$Number
$.mp4?m=1528475245" initialization="index_video_1_0_init.mp4?m=1528475245"
startNumber="178444" presentationTimeOffset="10395907501">
        <SegmentTimeline>
          <S t="10395907501" d="60060" r="29"/>
          <S t="10397709301" d="45045"/>

```

```

        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet mimeType="audio/mp4" segmentAlignment="0" lang="eng">
    <Representation id="2" bandwidth="96964" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <SegmentTemplate timescale="48000" media="index_audio_2_0_$Number
$.mp4?m=1528475245" initialization="index_audio_2_0_init.mp4?m=1528475245"
startNumber="178444" presentationTimeOffset="16633452001">
            <SegmentTimeline>
                <S t="16633452289" d="96256" r="3"/>
                <S t="16633837313" d="95232"/>
                <S t="16633932545" d="96256" r="4"/>
                <S t="16634413825" d="95232"/>
                <S t="16634509057" d="96256" r="5"/>
                <S t="16635086593" d="95232"/>
                <S t="16635181825" d="96256" r="4"/>
                <S t="16635663105" d="95232"/>
                <S t="16635758337" d="96256" r="5"/>
                <S t="16636335873" d="71680"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
</Period>

```

Beispiel einer personalisierten DASH-Antwort für Zeitsignal-Marker

AWS Elemental MediaTailor personalisiert die Werbeanzeige mit Werbespezifikationen. Die Personalisierungen spiegeln die vom Player empfangenen Daten zum Betrachter wider sowie die Werbekampagnen, die derzeit ausgeführt werden.

Das folgende Beispiel zeigt eine Anzeige, die nach AWS Elemental MediaTailor der Personalisierung verfügbar ist.

```

<Period id="178443_1" start="PT96H15M30.25S">
    <BaseURL>http://111122223333.cloudfront.net/nbc_fallback_2/</BaseURL>
    <AdaptationSet bitstreamSwitching="false" frameRate="30/1" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
        <SegmentTemplate startNumber="1" timescale="90000"/>
    </AdaptationSet>
</Period>

```

```

    <Representation bandwidth="10000000" codecs="avc1.640028" height="1080"
id="1" width="1920">
    <SegmentTemplate initialization="nbc_fallback_ad_2_1080p_10init.mp4"
media="nbc_fallback_ad_2_1080p_10_${Number%09d$.mp4" startNumber="1" timescale="90000">
    <SegmentTimeline>
    <S d="180000" r="13" t="0"/>
    <S d="176940" t="2520000"/>
    </SegmentTimeline>
    </SegmentTemplate>
    </Representation>
    <Representation bandwidth="4000000" codecs="avc1.64001f" height="720"
id="2" width="1280">
    <SegmentTemplate initialization="nbc_fallback_ad_2_720p_9init.mp4"
media="nbc_fallback_ad_2_720p_9_${Number%09d$.mp4" startNumber="1" timescale="90000">
    <SegmentTimeline>
    <S d="180000" r="13" t="0"/>
    <S d="176940" t="2520000"/>
    </SegmentTimeline>
    </SegmentTemplate>
    </Representation>
    <Representation bandwidth="2500000" codecs="avc1.64001f" height="720"
id="3" width="1280">
    <SegmentTemplate initialization="nbc_fallback_ad_2_720p_8init.mp4"
media="nbc_fallback_ad_2_720p_8_${Number%09d$.mp4" startNumber="1" timescale="90000">
    <SegmentTimeline>
    <S d="270000" r="8" t="0"/>
    <S d="266940" t="2430000"/>
    </SegmentTimeline>
    </SegmentTemplate>
    </Representation>
    <Representation bandwidth="2000000" codecs="avc1.64001f" height="540"
id="4" width="960">
    <SegmentTemplate initialization="nbc_fallback_ad_2_540p_7init.mp4"
media="nbc_fallback_ad_2_540p_7_${Number%09d$.mp4" startNumber="1" timescale="90000">
    <SegmentTimeline>
    <S d="360000" r="6" t="0"/>
    <S d="176940" t="2520000"/>
    </SegmentTimeline>
    </SegmentTemplate>
    </Representation>
    <Representation bandwidth="1350000" codecs="avc1.64001e" height="396"
id="5" width="704">
    <SegmentTemplate initialization="nbc_fallback_ad_2_396p_6init.mp4"
media="nbc_fallback_ad_2_396p_6_${Number%09d$.mp4" startNumber="1" timescale="90000">

```



```

        <SegmentTimeline>
          <S d="360000" r="6" t="0"/>
          <S d="176940" t="2520000"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="900000" codecs="avc1.64001e" height="396" id="6"
width="704">
      <SegmentTemplate initialization="nbc_fallback_ad_2_396p_5init.mp4"
media="nbc_fallback_ad_2_396p_5_$Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="360000" r="6" t="0"/>
          <S d="176940" t="2520000"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="600000" codecs="avc1.64001e" height="396" id="7"
width="704">
      <SegmentTemplate initialization="nbc_fallback_ad_2_396p_4init.mp4"
media="nbc_fallback_ad_2_396p_4_$Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="360000" r="6" t="0"/>
          <S d="176940" t="2520000"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="450000" codecs="avc1.640016" height="288" id="8"
width="512">
      <SegmentTemplate initialization="nbc_fallback_ad_2_288p_3init.mp4"
media="nbc_fallback_ad_2_288p_3_$Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="360000" r="6" t="0"/>
          <S d="176940" t="2520000"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="300000" codecs="avc1.640016" height="288" id="9"
width="512">
      <SegmentTemplate initialization="nbc_fallback_ad_2_288p_2init.mp4"
media="nbc_fallback_ad_2_288p_2_$Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="360000" r="6" t="0"/>
          <S d="176940" t="2520000"/>
        </SegmentTimeline>

```

```

        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="200000" codecs="avc1.640016" height="288"
id="10" width="512">
        <SegmentTemplate initialization="nbc_fallback_ad_2_288p_1init.mp4"
media="nbc_fallback_ad_2_288p_1_${Number%09d$.mp4" startNumber="1" timescale="90000">
            <SegmentTimeline>
                <S d="180000" r="13" t="0"/>
                <S d="176940" t="2520000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate
initialization="nbc_fallback_ad_2_audio_aac_a1_128kinit.mp4"
media="nbc_fallback_ad_2_audio_aac_a1_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"/>
        <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="11">
            <SegmentTemplate
initialization="nbc_fallback_ad_2_audio_aac_a1_128kinit.mp4"
media="nbc_fallback_ad_2_audio_aac_a1_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000">
                <SegmentTimeline>
                    <S d="96000" r="13" t="0"/>
                    <S d="94368" t="1344000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>
<AdaptationSet lang="enm" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate
initialization="nbc_fallback_ad_2_audio_aac_a2_128kinit.mp4"
media="nbc_fallback_ad_2_audio_aac_a2_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"/>
        <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="12">
            <SegmentTemplate
initialization="nbc_fallback_ad_2_audio_aac_a2_128kinit.mp4"
media="nbc_fallback_ad_2_audio_aac_a2_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000">
                <SegmentTimeline>
                    <S d="96000" r="13" t="0"/>

```

```

        <S d="94368" t="1344000"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet lang="por" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate
initialization="nbc_fallback_ad_2_audio_aac_a3_128kinit.mp4"
media="nbc_fallback_ad_2_audio_aac_a3_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"/>
        <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="13">
            <SegmentTemplate
initialization="nbc_fallback_ad_2_audio_aac_a3_128kinit.mp4"
media="nbc_fallback_ad_2_audio_aac_a3_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000">
                <SegmentTimeline>
                    <S d="96000" r="13" t="0"/>
                    <S d="94368" t="1344000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>
    <AdaptationSet lang="spa" mimeType="audio/mp4" segmentAlignment="0">
        <SegmentTemplate
initialization="nbc_fallback_ad_2_audio_aac_a4_128kinit.mp4"
media="nbc_fallback_ad_2_audio_aac_a4_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"/>
            <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="14">
                <SegmentTemplate
initialization="nbc_fallback_ad_2_audio_aac_a4_128kinit.mp4"
media="nbc_fallback_ad_2_audio_aac_a4_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000">
                    <SegmentTimeline>
                        <S d="96000" r="13" t="0"/>
                        <S d="94368" t="1344000"/>
                    </SegmentTimeline>
                </SegmentTemplate>
            </Representation>
        </AdaptationSet>
    </Period>

```

Base64-kodiertes Binärbeispiel für das DASH-Manifest mit einperiodischer Eingabe

Dieses Beispiel zeigt, wie AWS Elemental MediaTailor mit einem Manifest von einem Ursprungsserver umgegangen wird, der Manifeste für einzelne Perioden erzeugt. In Ihren MediaTailor Konfigurationseinstellungen können Sie angeben, dass Ihr Ursprungsserver Manifeste für einzelne Perioden erzeugt. MediaTailor erzeugt DASH-Manifeste mit mehreren Perioden, sowohl für Eingabemanifeste mit mehreren Perioden als auch für Einperioden.

Beispiel eines DASH-Manifests mit einem einzigen Zeitraum für das base64-kodierte binäre Format

Das folgende Beispiel zeigt den `<EventStream>` des Eingabezeitraums mit base64-kodierten binären Ad-Avail-Ereignissen.

```
<Period id="1" start="PT0S">
  <BaseURL>dash/</BaseURL>
  <EventStream schemeIdUri="urn:scte:scte35:2014:xml+bin" timescale="1">
    <Event presentationTime="1550252760" duration="24" id="136">
      <Signal xmlns="http://www.scte.org/schemas/35/2016">
        <Binary>/DAhAAAAAAAAAP/wEAUAAACIf+9/fgAg9YDAAAAAABiJjIs</
Binary>
        </Signal>
      </Event>
    <Event presentationTime="1550252880" duration="24" id="137">
      <Signal xmlns="http://www.scte.org/schemas/35/2016">
        <Binary>/DAhAAAAAAAAAP/wEAUAAACJf+9/fgAg9YDAAAAAAC/KdNe</
Binary>
        </Signal>
      </Event>
    <Event presentationTime="1550253000" duration="24" id="138">
      <Signal xmlns="http://www.scte.org/schemas/35/2016">
        <Binary>/DAhAAAAAAAAAP/wEAUAAACKf+9/fgAg9YDAAAAAADc+01/</
Binary>
        </Signal>
      </Event>
    </EventStream>
    <AdaptationSet...
  </AdaptationSet>
</Period>
```

Beispiel einer personalisierten DASH-Antwort für das base64-kodierte binäre Format mit der Ursprungs-Manifest-Konfiguration mit einem einzigen Zeitraum

Das folgende Beispiel spiegelt die Personalisierung wider, die auf die vorherige Anzeige angewendet wurde AWS Elemental MediaTailor , wenn die MediaTailor Konfiguration DASH-Manifeste für einzelne Perioden vom Ursprungsserver angibt. MediaTailor erzeugt ein DASH-Manifest für mehrere Perioden mit Personalisierungen, die die vom Player erhaltenen Zuschauerdaten und die aktuell laufenden Werbekampagnen widerspiegeln.

```
<Period id="0.0" start="PT0S">
  <BaseURL>dash/</BaseURL>
  <AdaptationSet audioSamplingRate="48000" codecs="mp4a.40.2"
contentType="audio" group="1" id="1" mimeType="audio/mp4" segmentAlignment="true"
startWithSAP="1">
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="1"/>
    <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
    <Representation bandwidth="69000" id="audio=69000">
      <SegmentTemplate initialization="scte35-$RepresentationID
$.dash" media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="0"
startNumber="1" timescale="48000">
        <SegmentTimeline>
          <S d="48129" t="74412130844415"/>
          <S d="48128" t="74412130892544"/>
          <S d="48127" t="74412130940672"/>
          <S d="48129" t="74412130988799"/>
          <S d="48128" t="74412131036928"/>
          <S d="47104" t="74412131085056"/>
          <S d="48128" t="74412131132160"/>
          <S d="48127" t="74412131180288"/>
          <S d="48129" t="74412131228415"/>
          <S d="48128" t="74412131276544"/>
          <S d="48127" t="74412131324672"/>
          <S d="48129" t="74412131372799"/>
          <S d="48128" t="74412131420928"/>
          <S d="47104" t="74412131469056"/>
          <S d="48128" t="74412131516160"/>
          <S d="48127" t="74412131564288"/>
          <S d="48129" t="74412131612415"/>
          <S d="48128" t="74412131660544"/>
          <S d="48127" t="74412131708672"/>
          <S d="48129" t="74412131756799"/>
          <S d="48128" t="74412131804928"/>
          <S d="47104" t="74412131853056"/>
          <S d="48128" t="74412131900160"/>
          <S d="48127" t="74412131948288"/>
        </SegmentTimeline>
      </Representation>
    </AdaptationSet>
  </Period>
```

```

        <S d="48129" t="74412131996415"/>
        <S d="48128" t="74412132044544"/>
        <S d="48127" t="74412132092672"/>
        <S d="48129" t="74412132140799"/>
        <S d="48128" t="74412132188928"/>
        <S d="47104" t="74412132237056"/>
        <S d="48128" t="74412132284160"/>
        <S d="48127" t="74412132332288"/>
        <S d="48129" t="74412132380415"/>
        <S d="48128" t="74412132428544"/>
        <S d="48127" t="74412132476672"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet codecs="avc1.64001F" contentType="video" group="2"
height="720" id="2" mimeType="video/mp4" par="16:9" sar="1:1" segmentAlignment="true"
startWithSAP="1" width="1280">
    <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
    <Representation bandwidth="700000" id="video=700000"
scanType="progressive">
        <SegmentTemplate initialization="scte35-$RepresentationID
$.dash" media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="0"
startNumber="1" timescale="90000">
            <SegmentTimeline>
                <S d="90000" r="34" t="139522745250000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
</Period>
<Period id="1550252760.0_1" start="PT430625H46M">
    <BaseURL>http://d2gh0tfpz97e4o.cloudfront.net/visitalps/</BaseURL>
    <AdaptationSet bitstreamSwitching="false" frameRate="30/1"
mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SegmentTemplate startNumber="1" timescale="90000"/>
        <Representation bandwidth="7500000" codecs="avc1.640028"
height="1080" id="1" width="1920">
            <SegmentTemplate
initialization="visitalps_1080p30_video_1080p_10init.mp4"
media="visitalps_1080p30_video_1080p_10_$Number%09d$.mp4" startNumber="1"
timescale="90000">
                <SegmentTimeline>

```

```

        <S d="180000" r="6" t="0"/>
        <S d="86940" t="1260000"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
<Representation bandwidth="3000000" codecs="avc1.64001f" height="720"
id="2" width="1280">
    <SegmentTemplate
initialization="visitalps_1080p30_video_720p_9init.mp4"
media="visitalps_1080p30_video_720p_9_${Number%09d$.mp4" startNumber="1"
timescale="90000">
        <SegmentTimeline>
            <S d="180000" r="6" t="0"/>
            <S d="86940" t="1260000"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="1875000" codecs="avc1.64001f" height="720"
id="3" width="1280">
    <SegmentTemplate
initialization="visitalps_1080p30_video_720p_8init.mp4"
media="visitalps_1080p30_video_720p_8_${Number%09d$.mp4" startNumber="1"
timescale="90000">
        <SegmentTimeline>
            <S d="270000" r="3" t="0"/>
            <S d="266940" t="1080000"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="1500000" codecs="avc1.64001f" height="540"
id="4" width="960">
    <SegmentTemplate
initialization="visitalps_1080p30_video_540p_7init.mp4"
media="visitalps_1080p30_video_540p_7_${Number%09d$.mp4" startNumber="1"
timescale="90000">
        <SegmentTimeline>
            <S d="360000" r="2" t="0"/>
            <S d="266940" t="1080000"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="1012500" codecs="avc1.64001e" height="396"
id="5" width="704">

```

```

        <SegmentTemplate
initialization="visitalps_1080p30_video_396p_6init.mp4"
media="visitalps_1080p30_video_396p_6_${Number%09d$.mp4" startNumber="1"
timescale="90000">
            <SegmentTimeline>
                <S d="360000" r="2" t="0"/>
                <S d="266940" t="1080000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="675000" codecs="avc1.64001e" height="396"
id="6" width="704">
        <SegmentTemplate
initialization="visitalps_1080p30_video_396p_5init.mp4"
media="visitalps_1080p30_video_396p_5_${Number%09d$.mp4" startNumber="1"
timescale="90000">
            <SegmentTimeline>
                <S d="360000" r="2" t="0"/>
                <S d="266940" t="1080000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="450000" codecs="avc1.64001e" height="396"
id="7" width="704">
        <SegmentTemplate
initialization="visitalps_1080p30_video_396p_4init.mp4"
media="visitalps_1080p30_video_396p_4_${Number%09d$.mp4" startNumber="1"
timescale="90000">
            <SegmentTimeline>
                <S d="360000" r="2" t="0"/>
                <S d="266940" t="1080000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="337500" codecs="avc1.640016" height="288"
id="8" width="512">
        <SegmentTemplate
initialization="visitalps_1080p30_video_288p_3init.mp4"
media="visitalps_1080p30_video_288p_3_${Number%09d$.mp4" startNumber="1"
timescale="90000">
            <SegmentTimeline>
                <S d="360000" r="2" t="0"/>
                <S d="266940" t="1080000"/>
            </SegmentTimeline>

```



```

        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="225000" codecs="avc1.640016" height="288"
id="9" width="512">
        <SegmentTemplate
initialization="visitalps_1080p30_video_288p_2init.mp4"
media="visitalps_1080p30_video_288p_2_${Number%09d$.mp4" startNumber="1"
timescale="90000">
            <SegmentTimeline>
                <S d="360000" r="2" t="0"/>
                <S d="266940" t="1080000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="150000" codecs="avc1.640016" height="288"
id="10" width="512">
        <SegmentTemplate
initialization="visitalps_1080p30_video_288p_1init.mp4"
media="visitalps_1080p30_video_288p_1_${Number%09d$.mp4" startNumber="1"
timescale="90000">
            <SegmentTimeline>
                <S d="180000" r="6" t="0"/>
                <S d="86940" t="1260000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate
initialization="visitalps_1080p30_audio_aac_128kinit.mp4"
media="visitalps_1080p30_audio_aac_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"/>
        <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="11">
            <SegmentTemplate
initialization="visitalps_1080p30_audio_aac_128kinit.mp4"
media="visitalps_1080p30_audio_aac_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000">
                <SegmentTimeline>
                    <S d="96000" r="6" t="0"/>
                    <S d="46368" t="672000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>

```

```

        </AdaptationSet>
    </Period>
    <Period id="1550252760.0" start="PT430625H46M14.966S">
        <BaseURL>dash/</BaseURL>
        <EventStream schemeIdUri="urn:scte:scte35:2014:xml+bin" timescale="1">
            <Event duration="24" id="136" presentationTime="1550252760">
                <Signal xmlns="http://www.scte.org/schemas/35/2016">
                    <Binary>/DAhAAAAAAAAAAP/wEAUAAACIf+9/fgAg9YDAAAAAAAAABiJjIs</
Binary>
                    </Signal>
                </Event>
            </EventStream>
            <AdaptationSet audioSamplingRate="48000" codecs="mp4a.40.2"
contentType="audio" group="1" id="1" mimeType="audio/mp4" segmentAlignment="true"
startWithSAP="1">
                <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="1"/>
                <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
                <Representation bandwidth="69000" id="audio=69000">
                    <SegmentTemplate initialization="scte35-$RepresentationID$.dash"
media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="74412133198368"
timescale="48000">
                        <SegmentTimeline>
                            <S d="48128" t="74412133196544"/>
                            <S d="48127" t="74412133244672"/>
                            <S d="48129" t="74412133292799"/>
                            <S d="48128" t="74412133340928"/>
                            <S d="47104" t="74412133389056"/>
                            <S d="48128" t="74412133436160"/>
                            <S d="48127" t="74412133484288"/>
                            <S d="48129" t="74412133532415"/>
                            <S d="48128" t="74412133580544"/>
                            <S d="48127" t="74412133628672"/>
                        </SegmentTimeline>
                    </SegmentTemplate>
                </Representation>
            </AdaptationSet>
            <AdaptationSet codecs="avc1.64001F" contentType="video" group="2"
height="720" id="2" mimeType="video/mp4" par="16:9" sar="1:1" segmentAlignment="true"
startWithSAP="1" width="1280">
                <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
                <Representation bandwidth="700000" id="video=700000"
scanType="progressive">

```

```

        <SegmentTemplate initialization="scte35-$RepresentationID$.dash"
media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="139522749746940"
timescale="90000">
            <SegmentTimeline>
                <S d="90000" r="9" t="139522749660000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
</Period>
<Period id="1550252784.0" start="PT430625H46M24S">
    <BaseURL>dash/</BaseURL>
    <AdaptationSet audioSamplingRate="48000" codecs="mp4a.40.2"
contentType="audio" group="1" id="1" mimeType="audio/mp4" segmentAlignment="true"
startWithSAP="1">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="1"/>
        <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
        <Representation bandwidth="69000" id="audio=69000">
            <SegmentTemplate initialization="scte35-$RepresentationID$.dash"
media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="74412133632000"
startNumber="60" timescale="48000">
                <SegmentTimeline>
                    <S d="48129" t="74412133676799"/>
                    <S d="48128" t="74412133724928"/>
                    <S d="47104" t="74412133773056"/>
                    <S d="48128" t="74412133820160"/>
                    <S d="48127" t="74412133868288"/>
                    <S d="48129" t="74412133916415"/>
                    <S d="48128" t="74412133964544"/>
                    <S d="48127" t="74412134012672"/>
                    <S d="48129" t="74412134060799"/>
                    <S d="48128" t="74412134108928"/>
                    <S d="47104" t="74412134157056"/>
                    <S d="48128" t="74412134204160"/>
                    <S d="48127" t="74412134252288"/>
                    <S d="48129" t="74412134300415"/>
                    <S d="48128" t="74412134348544"/>
                    <S d="48127" t="74412134396672"/>
                    <S d="48129" t="74412134444799"/>
                    <S d="48128" t="74412134492928"/>
                    <S d="47104" t="74412134541056"/>
                    <S d="48128" t="74412134588160"/>
                    <S d="48127" t="74412134636288"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>
</Period>
</Period>

```

```
<S d="48129" t="74412134684415"/>
<S d="48128" t="74412134732544"/>
<S d="48127" t="74412134780672"/>
<S d="48129" t="74412134828799"/>
<S d="48128" t="74412134876928"/>
<S d="47104" t="74412134925056"/>
<S d="48128" t="74412134972160"/>
<S d="48127" t="74412135020288"/>
<S d="48129" t="74412135068415"/>
<S d="48128" t="74412135116544"/>
<S d="48127" t="74412135164672"/>
<S d="48129" t="74412135212799"/>
<S d="48128" t="74412135260928"/>
<S d="47104" t="74412135309056"/>
<S d="48128" t="74412135356160"/>
<S d="48127" t="74412135404288"/>
<S d="48129" t="74412135452415"/>
<S d="48128" t="74412135500544"/>
<S d="48127" t="74412135548672"/>
<S d="48129" t="74412135596799"/>
<S d="48128" t="74412135644928"/>
<S d="47104" t="74412135693056"/>
<S d="48128" t="74412135740160"/>
<S d="48127" t="74412135788288"/>
<S d="48129" t="74412135836415"/>
<S d="48128" t="74412135884544"/>
<S d="48127" t="74412135932672"/>
<S d="48129" t="74412135980799"/>
<S d="48128" t="74412136028928"/>
<S d="47104" t="74412136077056"/>
<S d="48128" t="74412136124160"/>
<S d="48127" t="74412136172288"/>
<S d="48129" t="74412136220415"/>
<S d="48128" t="74412136268544"/>
<S d="48127" t="74412136316672"/>
<S d="48129" t="74412136364799"/>
<S d="48128" t="74412136412928"/>
<S d="47104" t="74412136461056"/>
<S d="48128" t="74412136508160"/>
<S d="48127" t="74412136556288"/>
<S d="48129" t="74412136604415"/>
<S d="48128" t="74412136652544"/>
<S d="48127" t="74412136700672"/>
<S d="48129" t="74412136748799"/>
```

```

        <S d="48128" t="74412136796928"/>
        <S d="47104" t="74412136845056"/>
        <S d="48128" t="74412136892160"/>
        <S d="48127" t="74412136940288"/>
        <S d="48129" t="74412136988415"/>
        <S d="48128" t="74412137036544"/>
        <S d="48127" t="74412137084672"/>
        <S d="48129" t="74412137132799"/>
        <S d="48128" t="74412137180928"/>
        <S d="47104" t="74412137229056"/>
        <S d="48128" t="74412137276160"/>
        <S d="48127" t="74412137324288"/>
        <S d="48129" t="74412137372415"/>
        <S d="48128" t="74412137420544"/>
        <S d="48127" t="74412137468672"/>
        <S d="48129" t="74412137516799"/>
        <S d="48128" t="74412137564928"/>
        <S d="47104" t="74412137613056"/>
        <S d="48128" t="74412137660160"/>
        <S d="48127" t="74412137708288"/>
        <S d="48129" t="74412137756415"/>
        <S d="48128" t="74412137804544"/>
        <S d="48127" t="74412137852672"/>
        <S d="48129" t="74412137900799"/>
        <S d="48128" t="74412137948928"/>
        <S d="47104" t="74412137997056"/>
        <S d="48128" t="74412138044160"/>
        <S d="48127" t="74412138092288"/>
        <S d="48129" t="74412138140415"/>
        <S d="48128" t="74412138188544"/>
        <S d="48127" t="74412138236672"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet codecs="avc1.64001F" contentType="video" group="2"
height="720" id="2" mimeType="video/mp4" par="16:9" sar="1:1" segmentAlignment="true"
startWithSAP="1" width="1280">
    <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
    <Representation bandwidth="700000" id="video=700000"
scanType="progressive">
        <SegmentTemplate initialization="scte35-$RepresentationID$.dash"
media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="139522750560000"
startNumber="60" timescale="90000">

```

```

        <SegmentTimeline>
            <S d="90000" r="95" t="139522750560000"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
</Period>
<Period id="1550252880.0_1" start="PT430625H48M">
    <BaseURL>http://d2gh0tfpz97e4o.cloudfront.net/visitalps/</BaseURL>
    <AdaptationSet bitstreamSwitching="false" frameRate="30/1"
mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SegmentTemplate startNumber="1" timescale="90000"/>
        <Representation bandwidth="7500000" codecs="avc1.640028"
height="1080" id="1" width="1920">
            <SegmentTemplate
initialization="visitalps_1080p30_video_1080p_10init.mp4"
media="visitalps_1080p30_video_1080p_10_${Number%09d$.mp4" startNumber="1"
timescale="90000">
                <SegmentTimeline>
                    <S d="180000" r="6" t="0"/>
                    <S d="86940" t="1260000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="3000000" codecs="avc1.64001f" height="720"
id="2" width="1280">
            <SegmentTemplate
initialization="visitalps_1080p30_video_720p_9init.mp4"
media="visitalps_1080p30_video_720p_9_${Number%09d$.mp4" startNumber="1"
timescale="90000">
                <SegmentTimeline>
                    <S d="180000" r="6" t="0"/>
                    <S d="86940" t="1260000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="1875000" codecs="avc1.64001f" height="720"
id="3" width="1280">
            <SegmentTemplate
initialization="visitalps_1080p30_video_720p_8init.mp4"
media="visitalps_1080p30_video_720p_8_${Number%09d$.mp4" startNumber="1"
timescale="90000">
                <SegmentTimeline>

```

```

        <S d="270000" r="3" t="0"/>
        <S d="266940" t="1080000"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
<Representation bandwidth="1500000" codecs="avc1.64001f" height="540"
id="4" width="960">
    <SegmentTemplate
initialization="visitalps_1080p30_video_540p_7init.mp4"
media="visitalps_1080p30_video_540p_7_${Number%09d$.mp4" startNumber="1"
timescale="90000">
        <SegmentTimeline>
            <S d="360000" r="2" t="0"/>
            <S d="266940" t="1080000"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="1012500" codecs="avc1.64001e" height="396"
id="5" width="704">
    <SegmentTemplate
initialization="visitalps_1080p30_video_396p_6init.mp4"
media="visitalps_1080p30_video_396p_6_${Number%09d$.mp4" startNumber="1"
timescale="90000">
        <SegmentTimeline>
            <S d="360000" r="2" t="0"/>
            <S d="266940" t="1080000"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="675000" codecs="avc1.64001e" height="396"
id="6" width="704">
    <SegmentTemplate
initialization="visitalps_1080p30_video_396p_5init.mp4"
media="visitalps_1080p30_video_396p_5_${Number%09d$.mp4" startNumber="1"
timescale="90000">
        <SegmentTimeline>
            <S d="360000" r="2" t="0"/>
            <S d="266940" t="1080000"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="450000" codecs="avc1.64001e" height="396"
id="7" width="704">

```

```

        <SegmentTemplate
initialization="visitalps_1080p30_video_396p_4init.mp4"
media="visitalps_1080p30_video_396p_4_${Number%09d$.mp4" startNumber="1"
timescale="90000">
            <SegmentTimeline>
                <S d="360000" r="2" t="0"/>
                <S d="266940" t="1080000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="337500" codecs="avc1.640016" height="288"
id="8" width="512">
        <SegmentTemplate
initialization="visitalps_1080p30_video_288p_3init.mp4"
media="visitalps_1080p30_video_288p_3_${Number%09d$.mp4" startNumber="1"
timescale="90000">
            <SegmentTimeline>
                <S d="360000" r="2" t="0"/>
                <S d="266940" t="1080000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="225000" codecs="avc1.640016" height="288"
id="9" width="512">
        <SegmentTemplate
initialization="visitalps_1080p30_video_288p_2init.mp4"
media="visitalps_1080p30_video_288p_2_${Number%09d$.mp4" startNumber="1"
timescale="90000">
            <SegmentTimeline>
                <S d="360000" r="2" t="0"/>
                <S d="266940" t="1080000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="150000" codecs="avc1.640016" height="288"
id="10" width="512">
        <SegmentTemplate
initialization="visitalps_1080p30_video_288p_1init.mp4"
media="visitalps_1080p30_video_288p_1_${Number%09d$.mp4" startNumber="1"
timescale="90000">
            <SegmentTimeline>
                <S d="180000" r="6" t="0"/>
                <S d="86940" t="1260000"/>
            </SegmentTimeline>

```



```

        </SegmentTemplate>
    </Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate
initialization="visitalps_1080p30_audio_aac_128kinit.mp4"
media="visitalps_1080p30_audio_aac_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"/>
        <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="11">
            <SegmentTemplate
initialization="visitalps_1080p30_audio_aac_128kinit.mp4"
media="visitalps_1080p30_audio_aac_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000">
                <SegmentTimeline>
                    <S d="96000" r="6" t="0"/>
                    <S d="46368" t="672000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>
</Period>
<Period id="1550252880.0" start="PT430625H48M14.966S">
    <BaseURL>dash</BaseURL>
    <EventStream schemeIdUri="urn:scte:scte35:2014:xml+bin" timescale="1">
        <Event duration="24" id="137" presentationTime="1550252880">
            <Signal xmlns="http://www.scte.org/schemas/35/2016">
                <Binary>/DAhAAAAAAAAAAP/wEAUAAACJf+9/fgAg9YDAAAAAAAAAC/KdNe</
Binary>
                </Signal>
            </Event>
        </EventStream>
    <AdaptationSet audioSamplingRate="48000" codecs="mp4a.40.2"
contentType="audio" group="1" id="1" mimeType="audio/mp4" segmentAlignment="true"
startWithSAP="1">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="1"/>
        <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
        <Representation bandwidth="69000" id="audio=69000">
            <SegmentTemplate initialization="scte35-$RepresentationID$.dash"
media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="74412138958368"
timescale="48000">
                <SegmentTimeline>
                    <S d="48128" t="74412138956544"/>

```

```

        <S d="48127" t="74412139004672"/>
        <S d="48129" t="74412139052799"/>
        <S d="48128" t="74412139100928"/>
        <S d="47104" t="74412139149056"/>
        <S d="48128" t="74412139196160"/>
        <S d="48127" t="74412139244288"/>
        <S d="48129" t="74412139292415"/>
        <S d="48128" t="74412139340544"/>
        <S d="48127" t="74412139388672"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet codecs="avc1.64001F" contentType="video" group="2"
height="720" id="2" mimeType="video/mp4" par="16:9" sar="1:1" segmentAlignment="true"
startWithSAP="1" width="1280">
    <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
    <Representation bandwidth="700000" id="video=700000"
scanType="progressive">
        <SegmentTemplate initialization="scte35-$RepresentationID$.dash"
media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="139522760546940"
timescale="90000">
            <SegmentTimeline>
                <S d="90000" r="9" t="139522760460000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
</Period>
<Period id="1550252904.0" start="PT430625H48M24S">
    <BaseURL>dash/</BaseURL>
    <AdaptationSet audioSamplingRate="48000" codecs="mp4a.40.2"
contentType="audio" group="1" id="1" mimeType="audio/mp4" segmentAlignment="true"
startWithSAP="1">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="1"/>
        <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
        <Representation bandwidth="69000" id="audio=69000">
            <SegmentTemplate initialization="scte35-$RepresentationID$.dash"
media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="74412139392000"
startNumber="180" timescale="48000">
                <SegmentTimeline>
                    <S d="48129" t="74412139436799"/>
                    <S d="48128" t="74412139484928"/>

```

```
<S d="47104" t="74412139533056"/>
<S d="48128" t="74412139580160"/>
<S d="48127" t="74412139628288"/>
<S d="48129" t="74412139676415"/>
<S d="48128" t="74412139724544"/>
<S d="48127" t="74412139772672"/>
<S d="48129" t="74412139820799"/>
<S d="48128" t="74412139868928"/>
<S d="47104" t="74412139917056"/>
<S d="48128" t="74412139964160"/>
<S d="48127" t="74412140012288"/>
<S d="48129" t="74412140060415"/>
<S d="48128" t="74412140108544"/>
<S d="48127" t="74412140156672"/>
<S d="48129" t="74412140204799"/>
<S d="48128" t="74412140252928"/>
<S d="47104" t="74412140301056"/>
<S d="48128" t="74412140348160"/>
<S d="48127" t="74412140396288"/>
<S d="48129" t="74412140444415"/>
<S d="48128" t="74412140492544"/>
<S d="48127" t="74412140540672"/>
<S d="48129" t="74412140588799"/>
<S d="48128" t="74412140636928"/>
<S d="47104" t="74412140685056"/>
<S d="48128" t="74412140732160"/>
<S d="48127" t="74412140780288"/>
<S d="48129" t="74412140828415"/>
<S d="48128" t="74412140876544"/>
<S d="48127" t="74412140924672"/>
<S d="48129" t="74412140972799"/>
<S d="48128" t="74412141020928"/>
<S d="47104" t="74412141069056"/>
<S d="48128" t="74412141116160"/>
<S d="48127" t="74412141164288"/>
<S d="48129" t="74412141212415"/>
<S d="48128" t="74412141260544"/>
<S d="48127" t="74412141308672"/>
<S d="48129" t="74412141356799"/>
<S d="48128" t="74412141404928"/>
<S d="47104" t="74412141453056"/>
<S d="48128" t="74412141500160"/>
<S d="48127" t="74412141548288"/>
<S d="48129" t="74412141596415"/>
```

```
<S d="48128" t="74412141644544"/>
<S d="48127" t="74412141692672"/>
<S d="48129" t="74412141740799"/>
<S d="48128" t="74412141788928"/>
<S d="47104" t="74412141837056"/>
<S d="48128" t="74412141884160"/>
<S d="48127" t="74412141932288"/>
<S d="48129" t="74412141980415"/>
<S d="48128" t="74412142028544"/>
<S d="48127" t="74412142076672"/>
<S d="48129" t="74412142124799"/>
<S d="48128" t="74412142172928"/>
<S d="47104" t="74412142221056"/>
<S d="48128" t="74412142268160"/>
<S d="48127" t="74412142316288"/>
<S d="48129" t="74412142364415"/>
<S d="48128" t="74412142412544"/>
<S d="48127" t="74412142460672"/>
<S d="48129" t="74412142508799"/>
<S d="48128" t="74412142556928"/>
<S d="47104" t="74412142605056"/>
<S d="48128" t="74412142652160"/>
<S d="48127" t="74412142700288"/>
<S d="48129" t="74412142748415"/>
<S d="48128" t="74412142796544"/>
<S d="48127" t="74412142844672"/>
<S d="48129" t="74412142892799"/>
<S d="48128" t="74412142940928"/>
<S d="47104" t="74412142989056"/>
<S d="48128" t="74412143036160"/>
<S d="48127" t="74412143084288"/>
<S d="48129" t="74412143132415"/>
<S d="48128" t="74412143180544"/>
<S d="48127" t="74412143228672"/>
<S d="48129" t="74412143276799"/>
<S d="48128" t="74412143324928"/>
<S d="47104" t="74412143373056"/>
<S d="48128" t="74412143420160"/>
<S d="48127" t="74412143468288"/>
<S d="48129" t="74412143516415"/>
<S d="48128" t="74412143564544"/>
<S d="48127" t="74412143612672"/>
<S d="48129" t="74412143660799"/>
<S d="48128" t="74412143708928"/>
```

```

        <S d="47104" t="74412143757056"/>
        <S d="48128" t="74412143804160"/>
        <S d="48127" t="74412143852288"/>
        <S d="48129" t="74412143900415"/>
        <S d="48128" t="74412143948544"/>
        <S d="48127" t="74412143996672"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet codecs="avc1.64001F" contentType="video" group="2"
height="720" id="2" mimeType="video/mp4" par="16:9" sar="1:1" segmentAlignment="true"
startWithSAP="1" width="1280">
    <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
    <Representation bandwidth="700000" id="video=700000"
scanType="progressive">
        <SegmentTemplate initialization="scte35-$RepresentationID$.dash"
media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="139522761360000"
startNumber="180" timescale="90000">
            <SegmentTimeline>
                <S d="90000" r="95" t="139522761360000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
</Period>
<Period id="1550253000.0_1" start="PT430625H50M">
    <BaseURL>http://d2gh0tfpz97e4o.cloudfront.net/visitalps</BaseURL>
    <AdaptationSet bitstreamSwitching="false" frameRate="30/1"
mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SegmentTemplate startNumber="1" timescale="90000"/>
        <Representation bandwidth="7500000" codecs="avc1.640028"
height="1080" id="1" width="1920">
            <SegmentTemplate
initialization="visitalps_1080p30_video_1080p_10init.mp4"
media="visitalps_1080p30_video_1080p_10_$Number%09d$.mp4" startNumber="1"
timescale="90000">
                <SegmentTimeline>
                    <S d="180000" r="6" t="0"/>
                    <S d="86940" t="1260000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>

```

```

        <Representation bandwidth="3000000" codecs="avc1.64001f" height="720"
id="2" width="1280">
            <SegmentTemplate
initialization="visitalps_1080p30_video_720p_9init.mp4"
media="visitalps_1080p30_video_720p_9_${Number%09d$.mp4" startNumber="1"
timescale="90000">
                <SegmentTimeline>
                    <S d="180000" r="6" t="0"/>
                    <S d="86940" t="1260000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="1875000" codecs="avc1.64001f" height="720"
id="3" width="1280">
            <SegmentTemplate
initialization="visitalps_1080p30_video_720p_8init.mp4"
media="visitalps_1080p30_video_720p_8_${Number%09d$.mp4" startNumber="1"
timescale="90000">
                <SegmentTimeline>
                    <S d="270000" r="3" t="0"/>
                    <S d="266940" t="1080000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="1500000" codecs="avc1.64001f" height="540"
id="4" width="960">
            <SegmentTemplate
initialization="visitalps_1080p30_video_540p_7init.mp4"
media="visitalps_1080p30_video_540p_7_${Number%09d$.mp4" startNumber="1"
timescale="90000">
                <SegmentTimeline>
                    <S d="360000" r="2" t="0"/>
                    <S d="266940" t="1080000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="1012500" codecs="avc1.64001e" height="396"
id="5" width="704">
            <SegmentTemplate
initialization="visitalps_1080p30_video_396p_6init.mp4"
media="visitalps_1080p30_video_396p_6_${Number%09d$.mp4" startNumber="1"
timescale="90000">
                <SegmentTimeline>
                    <S d="360000" r="2" t="0"/>

```

```

        <S d="266940" t="1080000"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
<Representation bandwidth="675000" codecs="avc1.64001e" height="396"
id="6" width="704">
    <SegmentTemplate
initialization="visitalps_1080p30_video_396p_5init.mp4"
media="visitalps_1080p30_video_396p_5_${Number%09d$.mp4" startNumber="1"
timescale="90000">
        <SegmentTimeline>
            <S d="360000" r="2" t="0"/>
            <S d="266940" t="1080000"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="450000" codecs="avc1.64001e" height="396"
id="7" width="704">
    <SegmentTemplate
initialization="visitalps_1080p30_video_396p_4init.mp4"
media="visitalps_1080p30_video_396p_4_${Number%09d$.mp4" startNumber="1"
timescale="90000">
        <SegmentTimeline>
            <S d="360000" r="2" t="0"/>
            <S d="266940" t="1080000"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="337500" codecs="avc1.640016" height="288"
id="8" width="512">
    <SegmentTemplate
initialization="visitalps_1080p30_video_288p_3init.mp4"
media="visitalps_1080p30_video_288p_3_${Number%09d$.mp4" startNumber="1"
timescale="90000">
        <SegmentTimeline>
            <S d="360000" r="2" t="0"/>
            <S d="266940" t="1080000"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="225000" codecs="avc1.640016" height="288"
id="9" width="512">
    <SegmentTemplate
initialization="visitalps_1080p30_video_288p_2init.mp4"

```

```

media="visitalps_1080p30_video_288p_2_${Number%09d$.mp4" startNumber="1"
timescale="90000">
    <SegmentTimeline>
        <S d="360000" r="2" t="0"/>
        <S d="266940" t="1080000"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
<Representation bandwidth="150000" codecs="avc1.640016" height="288"
id="10" width="512">
    <SegmentTemplate
initialization="visitalps_1080p30_video_288p_1init.mp4"
media="visitalps_1080p30_video_288p_1_${Number%09d$.mp4" startNumber="1"
timescale="90000">
        <SegmentTimeline>
            <S d="180000" r="6" t="0"/>
            <S d="86940" t="1260000"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate
initialization="visitalps_1080p30_audio_aac_128kinit.mp4"
media="visitalps_1080p30_audio_aac_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000"/>
        <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="11">
            <SegmentTemplate
initialization="visitalps_1080p30_audio_aac_128kinit.mp4"
media="visitalps_1080p30_audio_aac_128k_${Number%09d$.mp4" startNumber="1"
timescale="48000">
                <SegmentTimeline>
                    <S d="96000" r="6" t="0"/>
                    <S d="46368" t="672000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>
</Period>
<Period id="1550253000.0" start="PT430625H50M14.966S">
    <BaseURL>dash/</BaseURL>
    <EventStream schemeIdUri="urn:scte:scte35:2014:xml+bin" timescale="1">
        <Event duration="24" id="138" presentationTime="1550253000">

```



```

        <Signal xmlns="http://www.scte.org/schemas/35/2016">
            <Binary>/DAhAAAAAAAAAAP/wEAUAAACKf+9/fgAg9YDAAAAAADc+01/</
Binary>

            </Signal>
        </Event>
    </EventStream>
    <AdaptationSet audioSamplingRate="48000" codecs="mp4a.40.2"
contentType="audio" group="1" id="1" mimeType="audio/mp4" segmentAlignment="true"
startWithSAP="1">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="1"/>
        <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
        <Representation bandwidth="69000" id="audio=69000">
            <SegmentTemplate initialization="scte35-$RepresentationID$.dash"
media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="74412144718368"
timescale="48000">
                <SegmentTimeline>
                    <S d="48128" t="74412144716544"/>
                    <S d="48127" t="74412144764672"/>
                    <S d="48129" t="74412144812799"/>
                    <S d="48128" t="74412144860928"/>
                    <S d="47104" t="74412144909056"/>
                    <S d="48128" t="74412144956160"/>
                    <S d="48127" t="74412145004288"/>
                    <S d="48129" t="74412145052415"/>
                    <S d="48128" t="74412145100544"/>
                    <S d="48127" t="74412145148672"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>
    <AdaptationSet codecs="avc1.64001F" contentType="video" group="2"
height="720" id="2" mimeType="video/mp4" par="16:9" sar="1:1" segmentAlignment="true"
startWithSAP="1" width="1280">
        <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
        <Representation bandwidth="700000" id="video=700000"
scanType="progressive">
            <SegmentTemplate initialization="scte35-$RepresentationID$.dash"
media="scte35-$RepresentationID$-$Time$.dash" presentationTimeOffset="139522771346940"
timescale="90000">
                <SegmentTimeline>
                    <S d="90000" r="9" t="139522771260000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>

```

```

        </Representation>
    </AdaptationSet>
</Period>

```

Beispiele für VOD-DASH-Manifeste

Dieser Abschnitt enthält Beispiele für VOD-DASH-Manifeste. In jedem Beispiel wird ein Manifest aufgeführt, das vom Ursprungsserver empfangen wurde. Danach MediaTailor wurde das Manifest mit Werbung personalisiert.

DASH-VOD-Ursprungsmanifest

Das folgende Beispiel aus einem MPD-Manifest zeigt den Ad-Avail-Zeitraum in einem VOD-Manifest, das über DASH vom Inhaltsursprung empfangen wurde. Dieses Beispiel verwendet die `scte35:SpliceInsert`-Marker, wobei `outOfNetworkIndicator` auf `true` eingestellt ist.

```

<?xml version="1.0" encoding="UTF-8"?>
<Period start="PT0.000S" id="8778696" duration="PT29.229S">
    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2018-07-27T09:35:44.011Z"/>
    <EventStream timescale="30000" schemeIdUri="urn:scte:scte35:2013:xml">
        <Event duration="0" presentationTime="1317997547283">
            <scte35:SpliceInfoSection protocolVersion="0" tier="4095">
                <scte35:SpliceInsert spliceEventId="1"
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
                    <scte35:Program>
                        <scte35:SpliceTime ptsTime="1317997547283"/>
                    </scte35:Program>
                    <scte35:BreakDuration autoReturn="true" duration="0"/>
                </scte35:SpliceInsert>
            </scte35:SpliceInfoSection>
        </Event>
    </EventStream>
    <AdaptationSet mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
subsegmentAlignment="true" subsegmentStartsWithSAP="1" bitstreamSwitching="true">
        <Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="2200000" codecs="avc1.640029">
            <SegmentTemplate timescale="30000" media="index_video_7_0_$Number
$.mp4?m=1566416213" initialization="index_video_7_0_init.mp4?m=1566416213"
startNumber="8778700" presentationTimeOffset="1317997547283">
                <SegmentTimeline>
                    <S t="1317997547283" d="180180" r="3"/>
                </SegmentTimeline>
            </Representation>
        </AdaptationSet>
    </Period>

```

```

        <S t="1317998268003" d="156156"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
<Representation id="2" width="1280" height="720" frameRate="30000/1001"
bandwidth="3299968" codecs="avc1.640029">
    <SegmentTemplate timescale="30000" media="index_video_10_0_$Number
$.mp4?m=1566416213" initialization="index_video_10_0_init.mp4?m=1566416213"
startNumber="8778700" presentationTimeOffset="1317997547283">
        <SegmentTimeline>
            <S t="1317997547283" d="180180" r="3"/>
            <S t="1317998268003" d="156156"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation id="3" width="640" height="360" frameRate="30000/1001"
bandwidth="800000" codecs="avc1.4D401E">
    <SegmentTemplate timescale="30000" media="index_video_28_0_$Number
$.mp4?m=1566416213" initialization="index_video_28_0_init.mp4?m=1566416213"
startNumber="8778700" presentationTimeOffset="1317997547283">
        <SegmentTimeline>
            <S t="1317997547283" d="180180" r="3"/>
            <S t="1317998268003" d="156156"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet mimeType="audio/mp4" segmentAlignment="0" lang="eng">
    <Label>eng</Label>
    <Representation id="4" bandwidth="96636" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate timescale="48000" media="index_audio_5_0_$Number
$.mp4?m=1566416213" initialization="index_audio_5_0_init.mp4?m=1566416213"
startNumber="8778700" presentationTimeOffset="2108796075909">
            <SegmentTimeline>
                <S t="2108796075909" d="288768"/>
                <S t="2108796364677" d="287744"/>
                <S t="2108796652421" d="288768"/>
                <S t="2108796941189" d="287744"/>
                <S t="2108797228933" d="249856"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>

```

```

    </Representation>
    <Representation id="5" bandwidth="96636" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate timescale="48000" media="index_audio_8_0_$Number
$.mp4?m=1566416213" initialization="index_audio_8_0_init.mp4?m=1566416213"
startNumber="8778700" presentationTimeOffset="2108796075909">
            <SegmentTimeline>
                <S t="2108796075909" d="288768"/>
                <S t="2108796364677" d="287744"/>
                <S t="2108796652421" d="288768"/>
                <S t="2108796941189" d="287744"/>
                <S t="2108797228933" d="249856"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation id="6" bandwidth="64643" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate timescale="48000" media="index_audio_26_0_$Number
$.mp4?m=1566416213" initialization="index_audio_26_0_init.mp4?m=1566416213"
startNumber="8778700" presentationTimeOffset="2108796075909">
            <SegmentTimeline>
                <S t="2108796075909" d="288768"/>
                <S t="2108796364677" d="287744"/>
                <S t="2108796652421" d="288768"/>
                <S t="2108796941189" d="287744"/>
                <S t="2108797228933" d="249856"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
<AdaptationSet mimeType="application/mp4" codecs="stpp" segmentAlignment="true"
startWithSAP="1" bitstreamSwitching="true" lang="eng">
    <Label>eng</Label>
    <Representation id="7" bandwidth="0">
        <SegmentTemplate timescale="90000" media="index_subtitles_4_0_$Number
$.mp4?m=1566416213" initialization="index_subtitles_4_0_init.mp4?m=1566416213"
startNumber="8778700" presentationTimeOffset="3953992641850">
            <SegmentTimeline>
                <S t="3953992641850" d="540540" r="3"/>
                <S t="3953994804010" d="468468"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>

```

```

        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
</Period>
<Period start="PT29.229S" id="8778704" duration="PT18.818S">
    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2018-07-27T09:36:13.240Z"/>
    <AdaptationSet mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
subsegmentAlignment="true" subsegmentStartsWithSAP="1" bitstreamSwitching="true">
        <Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="2200000" codecs="avc1.640029">
            <SegmentTemplate timescale="30000" media="index_video_7_0_$Number
$.mp4?m=1566416213" initialization="index_video_7_0_init.mp4?m=1566416213"
startNumber="8778705" presentationTimeOffset="1317998424159">
                <SegmentTimeline>
                    <S t="1317998424159" d="24024"/>
                    <S t="1317998448183" d="180180" r="2"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation id="2" width="1280" height="720" frameRate="30000/1001"
bandwidth="3299968" codecs="avc1.640029">
            <SegmentTemplate timescale="30000" media="index_video_10_0_$Number
$.mp4?m=1566416213" initialization="index_video_10_0_init.mp4?m=1566416213"
startNumber="8778705" presentationTimeOffset="1317998424159">
                <SegmentTimeline>
                    <S t="1317998424159" d="24024"/>
                    <S t="1317998448183" d="180180" r="2"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation id="3" width="640" height="360" frameRate="30000/1001"
bandwidth="800000" codecs="avc1.4D401E">
            <SegmentTemplate timescale="30000" media="index_video_28_0_$Number
$.mp4?m=1566416213" initialization="index_video_28_0_init.mp4?m=1566416213"
startNumber="8778705" presentationTimeOffset="1317998424159">
                <SegmentTimeline>
                    <S t="1317998424159" d="24024"/>
                    <S t="1317998448183" d="180180" r="2"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>

```

```

    <AdaptationSet mimeType="audio/mp4" segmentAlignment="0" lang="eng">
      <Label>eng</Label>
      <Representation id="4" bandwidth="96636" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate timescale="48000" media="index_audio_5_0_$Number
$.mp4?m=1566416213" initialization="index_audio_5_0_init.mp4?m=1566416213"
startNumber="8778705" presentationTimeOffset="2108797478789">
          <SegmentTimeline>
            <S t="2108797478789" d="38912"/>
            <S t="2108797517701" d="287744"/>
            <S t="2108797805445" d="288768"/>
            <S t="2108798094213" d="287744"/>
          </SegmentTimeline>
        </SegmentTemplate>
      </Representation>
      <Representation id="5" bandwidth="96636" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate timescale="48000" media="index_audio_8_0_$Number
$.mp4?m=1566416213" initialization="index_audio_8_0_init.mp4?m=1566416213"
startNumber="8778705" presentationTimeOffset="2108797478789">
          <SegmentTimeline>
            <S t="2108797478789" d="38912"/>
            <S t="2108797517701" d="287744"/>
            <S t="2108797805445" d="288768"/>
            <S t="2108798094213" d="287744"/>
          </SegmentTimeline>
        </SegmentTemplate>
      </Representation>
      <Representation id="6" bandwidth="64643" audioSamplingRate="48000"
codecs="mp4a.40

```

DASH-VOD-Manifest mit personalisierter Antwort

Das folgende Beispiel spiegelt die Personalisierung wider, die MediaTailor für das Ursprungsmanifest gilt.

```

<?xml version="1.0" encoding="UTF-8"?>
<MPD id="201" minBufferTime="PT30S" profiles="urn:mpeg:dash:profile:isoff-
main:2011" type="static" xmlns="urn:mpeg:dash:schema:mpd:2011"

```

```

xmlns:scte35="urn:scte:scte35:2013:xml" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd">
  <BaseURL>https://444455556666.mediapackage.us-west-2.amazonaws.com/out/
v1/5f6a2197815e444a967f0c12f8325a11/</BaseURL>
  <Period duration="PT14.976S" id="8778696_PT0S_0" start="PT0S">
    <BaseURL>https://111122223333.mediatailor.us-west-2.amazonaws.com/v1/
dashsegment/0d598fad40f42c4644d1c5b7674438772ee23b12/dash-vod-insertion/a5a7cf24-
ee56-40e9-a0a2-82b483cf8650/8778696_PT0S/8778696_PT0S_0/</BaseURL>
    <AdaptationSet bitstreamSwitching="false" frameRate="30/1" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
      <SegmentTemplate startNumber="1" timescale="90000"/>
      <Representation bandwidth="3296000" codecs="avc1.64001f" height="720" id="1"
width="1280">
        <SegmentTemplate initialization="asset_720_3_1init.mp4" media="asset_720_3_1_
$Number%09d$.mp4" startNumber="1">
          <SegmentTimeline>
            <S d="180000" r="6" t="0"/>
            <S d="87000" t="1260000"/>
          </SegmentTimeline>
        </SegmentTemplate>
      </Representation>
      <Representation bandwidth="2200000" codecs="avc1.64001f" height="540" id="2"
width="960">
        <SegmentTemplate initialization="asset_540_2_0init.mp4" media="asset_540_2_0_
$Number%09d$.mp4" startNumber="1">
          <SegmentTimeline>
            <S d="180000" r="6" t="0"/>
            <S d="87000" t="1260000"/>
          </SegmentTimeline>
        </SegmentTemplate>
      </Representation>
      <Representation bandwidth="800000" codecs="avc1.64001e" height="360" id="3"
width="640">
        <SegmentTemplate initialization="asset_360_0_2init.mp4" media="asset_360_0_2_
$Number%09d$.mp4" startNumber="1">
          <SegmentTimeline>
            <S d="180000" r="6" t="0"/>
            <S d="87000" t="1260000"/>
          </SegmentTimeline>
        </SegmentTemplate>
      </Representation>
    </AdaptationSet>

```

```

    <AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
      <SegmentTemplate initialization="asset_audio_96_3init.mp4"
media="asset_audio_96_3_${Number%09d$.mp4" startNumber="1" timescale="48000"/>
      <Label>eng</Label>
      <Representation audioSamplingRate="48000" bandwidth="96000" codecs="mp4a.40.2"
id="4">
        <SegmentTemplate initialization="asset_audio_96_3init.mp4"
media="asset_audio_96_3_${Number%09d$.mp4" startNumber="1">
          <SegmentTimeline>
            <S d="96256" r="3" t="0"/>
            <S d="95232" t="385024"/>
            <S d="96256" r="1" t="480256"/>
            <S d="46080" t="672768"/>
          </SegmentTimeline>
        </SegmentTemplate>
      </Representation>
    </AdaptationSet>
    <AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
      <SegmentTemplate initialization="asset_audio_96_4init.mp4"
media="asset_audio_96_4_${Number%09d$.mp4" startNumber="1" timescale="48000"/>
      <Label>eng</Label>
      <Representation audioSamplingRate="48000" bandwidth="96000" codecs="mp4a.40.2"
id="5">
        <SegmentTemplate initialization="asset_audio_96_4init.mp4"
media="asset_audio_96_4_${Number%09d$.mp4" startNumber="1">
          <SegmentTimeline>
            <S d="96256" r="3" t="0"/>
            <S d="95232" t="385024"/>
            <S d="96256" r="1" t="480256"/>
            <S d="46080" t="672768"/>
          </SegmentTimeline>
        </SegmentTemplate>
      </Representation>
    </AdaptationSet>
  </Period>
  <Period duration="PT14.976S" id="8778696_PT0S_1" start="PT14.976S">
    <BaseURL>https://123456789012.mediataylor.us-west-2.amazonaws.com/v1/
dashsegment/0d598fad40f42c4644d1c5b7674438772ee23b12/dash-vod-insertion/a5a7cf24-
ee56-40e9-a0a2-82b483cf8650/8778696_PT0S/8778696_PT0S_1/</BaseURL>
    <AdaptationSet bitstreamSwitching="false" frameRate="30/1" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
      <SegmentTemplate startNumber="1" timescale="90000"/>

```



```

    <Representation bandwidth="3296000" codecs="avc1.64001f" height="720" id="1"
width="1280">
      <SegmentTemplate initialization="asset_720_3_1init.mp4" media="asset_720_3_1_
$Number%09d$.mp4" startNumber="1">
        <SegmentTimeline>
          <S d="180000" r="6" t="0"/>
          <S d="87000" t="1260000"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="2200000" codecs="avc1.64001f" height="540" id="2"
width="960">
      <SegmentTemplate initialization="asset_540_2_0init.mp4" media="asset_540_2_0_
$Number%09d$.mp4" startNumber="1">
        <SegmentTimeline>
          <S d="180000" r="6" t="0"/>
          <S d="87000" t="1260000"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="800000" codecs="avc1.64001e" height="360" id="3"
width="640">
      <SegmentTemplate initialization="asset_360_0_2init.mp4" media="asset_360_0_2_
$Number%09d$.mp4" startNumber="1">
        <SegmentTimeline>
          <S d="180000" r="6" t="0"/>
          <S d="87000" t="1260000"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
  <AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate initialization="asset_audio_96_3init.mp4"
media="asset_audio_96_3_$Number%09d$.mp4" startNumber="1" timescale="48000"/>
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="96000" codecs="mp4a.40.2"
id="4">
      <SegmentTemplate initialization="asset_audio_96_3init.mp4"
media="asset_audio_96_3_$Number%09d$.mp4" startNumber="1">
        <SegmentTimeline>
          <S d="96256" r="3" t="0"/>
          <S d="95232" t="385024"/>
          <S d="96256" r="1" t="480256"/>
          <S d="46080" t="672768"/>

```

```

        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate initialization="asset_audio_96_4init.mp4"
media="asset_audio_96_4_${Number%09d$.mp4" startNumber="1" timescale="48000"/>
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="96000" codecs="mp4a.40.2"
id="5">
        <SegmentTemplate initialization="asset_audio_96_4init.mp4"
media="asset_audio_96_4_${Number%09d$.mp4" startNumber="1">
            <SegmentTimeline>
                <S d="96256" r="3" t="0"/>
                <S d="95232" t="385024"/>
                <S d="96256" r="1" t="480256"/>
                <S d="46080" t="672768"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
</Period>
<Period duration="PT24.024S" id="8778696_PT29.952S" start="PT29.952S">
    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2018-07-27T09:35:44.011Z"/>
    <AdaptationSet bitstreamSwitching="true" mimeType="video/mp4"
segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
        <Representation bandwidth="2200000" codecs="avc1.640029" frameRate="30000/1001"
height="540" id="1" width="960">
            <SegmentTemplate initialization="index_video_7_0_init.mp4?
m=1566416213" media="index_video_7_0_${Number$.mp4?m=1566416213"
presentationTimeOffset="1317997547283" startNumber="8778700" timescale="30000">
                <SegmentTimeline>
                    <S d="180180" r="3" t="1317997547283"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="3299968" codecs="avc1.640029" frameRate="30000/1001"
height="720" id="2" width="1280">
            <SegmentTemplate initialization="index_video_10_0_init.mp4?
m=1566416213" media="index_video_10_0_${Number$.mp4?m=1566416213"
presentationTimeOffset="1317997547283" startNumber="8778700" timescale="30000">
                <SegmentTimeline>

```

```

        <S d="180180" r="3" t="1317997547283"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
<Representation bandwidth="800000" codecs="avc1.4D401E" frameRate="30000/1001"
height="360" id="3" width="640">
    <SegmentTemplate initialization="index_video_28_0_init.mp4?
m=1566416213" media="index_video_28_0_$.mp4?m=1566416213"
presentationTimeOffset="1317997547283" startNumber="8778700" timescale="30000">
        <SegmentTimeline>
            <S d="180180" r="3" t="1317997547283"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="96636" codecs="mp4a.40.2"
id="4">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate initialization="index_audio_5_0_init.mp4?
m=1566416213" media="index_audio_5_0_$.mp4?m=1566416213"
presentationTimeOffset="2108796075909" startNumber="8778700" timescale="48000">
            <SegmentTimeline>
                <S d="288768" t="2108796075909"/>
                <S d="287744" t="2108796364677"/>
                <S d="288768" t="2108796652421"/>
                <S d="287744" t="2108796941189"/>
                <S d="249856" t="2108797228933"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation audioSamplingRate="48000" bandwidth="96636" codecs="mp4a.40.2"
id="5">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate initialization="index_audio_8_0_init.mp4?
m=1566416213" media="index_audio_8_0_$.mp4?m=1566416213"
presentationTimeOffset="2108796075909" startNumber="8778700" timescale="48000">
            <SegmentTimeline>
                <S d="288768" t="2108796075909"/>
                <S d="287744" t="2108796364677"/>

```

```

        <S d="288768" t="2108796652421"/>
        <S d="287744" t="2108796941189"/>
        <S d="249856" t="2108797228933"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
<Representation audioSamplingRate="48000" bandwidth="64643" codecs="mp4a.40.2"
id="6">
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    <SegmentTemplate initialization="index_audio_26_0_init.mp4?
m=1566416213" media="index_audio_26_0_$.mp4?m=1566416213"
presentationTimeOffset="2108796075909" startNumber="8778700" timescale="48000">
        <SegmentTimeline>
            <S d="288768" t="2108796075909"/>
            <S d="287744" t="2108796364677"/>
            <S d="288768" t="2108796652421"/>
            <S d="287744" t="2108796941189"/>
            <S d="249856" t="2108797228933"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet bitstreamSwitching="true" codecs="stpp" lang="eng"
mimeType="application/mp4" segmentAlignment="true" startWithSAP="1">
    <Label>eng</Label>
    <Representation bandwidth="0" id="7">
        <SegmentTemplate initialization="index_subtitles_4_0_init.mp4?
m=1566416213" media="index_subtitles_4_0_$.mp4?m=1566416213"
presentationTimeOffset="3953992641850" startNumber="8778700" timescale="90000">
            <SegmentTimeline>
                <S d="540540" r="3" t="3953992641850"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
</Period>
<Period duration="PT14.976S" id="8778696_PT25S_0" start="PT53.976S">
    <BaseURL>https://123456789012.mediataylor.us-west-2.amazonaws.com/v1/
dashsegment/0d598fad40f42c4644d1c5b7674438772ee23b12/dash-vod-insertion/a5a7cf24-
ee56-40e9-a0a2-82b483cf8650/8778696_PT25S/8778696_PT25S_0/</BaseURL>
    <AdaptationSet bitstreamSwitching="false" frameRate="30/1" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">

```

```

    <SegmentTemplate startNumber="1" timescale="90000"/>
    <Representation bandwidth="3296000" codecs="avc1.64001f" height="720" id="1"
width="1280">
        <SegmentTemplate initialization="asset_720_3_1init.mp4" media="asset_720_3_1_
$Number%09d$.mp4" startNumber="1">
            <SegmentTimeline>
                <S d="180000" r="6" t="0"/>
                <S d="87000" t="1260000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="2200000" codecs="avc1.64001f" height="540" id="2"
width="960">
        <SegmentTemplate initialization="asset_540_2_0init.mp4" media="asset_540_2_0_
$Number%09d$.mp4" startNumber="1">
            <SegmentTimeline>
                <S d="180000" r="6" t="0"/>
                <S d="87000" t="1260000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="800000" codecs="avc1.64001e" height="360" id="3"
width="640">
        <SegmentTemplate initialization="asset_360_0_2init.mp4" media="asset_360_0_2_
$Number%09d$.mp4" startNumber="1">
            <SegmentTimeline>
                <S d="180000" r="6" t="0"/>
                <S d="87000" t="1260000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate initialization="asset_audio_96_3init.mp4"
media="asset_audio_96_3_$Number%09d$.mp4" startNumber="1" timescale="48000"/>
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="96000" codecs="mp4a.40.2"
id="4">
        <SegmentTemplate initialization="asset_audio_96_3init.mp4"
media="asset_audio_96_3_$Number%09d$.mp4" startNumber="1">
            <SegmentTimeline>
                <S d="96256" r="3" t="0"/>
                <S d="95232" t="385024"/>
                <S d="96256" r="1" t="480256"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>

```

```

        <S d="46080" t="672768"/>
    </SegmentTimeline>
</SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate initialization="asset_audio_96_4init.mp4"
media="asset_audio_96_4_${Number%09d$.mp4" startNumber="1" timescale="48000"/>
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="96000" codecs="mp4a.40.2"
id="5">
        <SegmentTemplate initialization="asset_audio_96_4init.mp4"
media="asset_audio_96_4_${Number%09d$.mp4" startNumber="1">
            <SegmentTimeline>
                <S d="96256" r="3" t="0"/>
                <S d="95232" t="385024"/>
                <S d="96256" r="1" t="480256"/>
                <S d="46080" t="672768"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
</Period>
<Period duration="PT14.976S" id="8778696_PT25S_1" start="PT1M8.952S">
    <BaseURL>https://123456789012.mediatailor.us-west-2.amazonaws.com/v1/
dashsegment/0d598fad40f42c4644d1c5b7674438772ee23b12/dash-vod-insertion/a5a7cf24-
ee56-40e9-a0a2-82b483cf8650/8778696_PT25S/8778696_PT25S_1/</BaseURL>
    <AdaptationSet bitstreamSwitching="false" frameRate="30/1" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
        <SegmentTemplate startNumber="1" timescale="90000"/>
        <Representation bandwidth="3296000" codecs="avc1.64001f" height="720" id="1"
width="1280">
            <SegmentTemplate initialization="asset_720_3_1init.mp4" media="asset_720_3_1_
${Number%09d$.mp4" startNumber="1">
                <SegmentTimeline>
                    <S d="180000" r="6" t="0"/>
                    <S d="87000" t="1260000"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="2200000" codecs="avc1.64001f" height="540" id="2"
width="960">

```

```

    <SegmentTemplate initialization="asset_540_2_0init.mp4" media="asset_540_2_0_
$Number%09d$.mp4" startNumber="1">
      <SegmentTimeline>
        <S d="180000" r="6" t="0"/>
        <S d="87000" t="1260000"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>
  <Representation bandwidth="800000" codecs="avc1.64001e" height="360" id="3"
width="640">
    <SegmentTemplate initialization="asset_360_0_2init.mp4" media="asset_360_0_2_
$Number%09d$.mp4" startNumber="1">
      <SegmentTimeline>
        <S d="180000" r="6" t="0"/>
        <S d="87000" t="1260000"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
  <SegmentTemplate initialization="asset_audio_96_3init.mp4"
media="asset_audio_96_3_$Number%09d$.mp4" startNumber="1" timescale="48000"/>
  <Label>eng</Label>
  <Representation audioSamplingRate="48000" bandwidth="96000" codecs="mp4a.40.2"
id="4">
    <SegmentTemplate initialization="asset_audio_96_3init.mp4"
media="asset_audio_96_3_$Number%09d$.mp4" startNumber="1">
      <SegmentTimeline>
        <S d="96256" r="3" t="0"/>
        <S d="95232" t="385024"/>
        <S d="96256" r="1" t="480256"/>
        <S d="46080" t="672768"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
  <SegmentTemplate initialization="asset_audio_96_4init.mp4"
media="asset_audio_96_4_$Number%09d$.mp4" startNumber="1" timescale="48000"/>
  <Label>eng</Label>
  <Representation audioSamplingRate="48000" bandwidth="96000" codecs="mp4a.40.2"
id="5">
    <SegmentTemplate initialization="asset_audio_96_4init.mp4"
media="asset_audio_96_4_$Number%09d$.mp4" startNumber="1">

```

```

        <SegmentTimeline>
          <S d="96256" r="3" t="0"/>
          <S d="95232" t="385024"/>
          <S d="96256" r="1" t="480256"/>
          <S d="46080" t="672768"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
</Period>
<Period duration="PT5.205S" id="8778696_PT1M23.928S" start="PT1M23.928S">
  <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2018-07-27T09:35:44.011Z"/>
  <AdaptationSet bitstreamSwitching="true" mimeType="video/mp4"
segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
    <Representation bandwidth="2200000" codecs="avc1.640029" frameRate="30000/1001"
height="540" id="1" width="960">
      <SegmentTemplate initialization="index_video_7_0_init.mp4?
m=1566416213" media="index_video_7_0_$.mp4?m=1566416213"
presentationTimeOffset="1317998268003" startNumber="8778704" timescale="30000">
        <SegmentTimeline>
          <S d="156156" t="1317998268003"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="3299968" codecs="avc1.640029" frameRate="30000/1001"
height="720" id="2" width="1280">
      <SegmentTemplate initialization="index_video_10_0_init.mp4?
m=1566416213" media="index_video_10_0_$.mp4?m=1566416213"
presentationTimeOffset="1317998268003" startNumber="8778704" timescale="30000">
        <SegmentTimeline>
          <S d="156156" t="1317998268003"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="800000" codecs="avc1.4D401E" frameRate="30000/1001"
height="360" id="3" width="640">
      <SegmentTemplate initialization="index_video_28_0_init.mp4?
m=1566416213" media="index_video_28_0_$.mp4?m=1566416213"
presentationTimeOffset="1317998268003" startNumber="8778704" timescale="30000">
        <SegmentTimeline>
          <S d="156156" t="1317998268003"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
</Period>

```



```

        </SegmentTemplate>
    </Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="96636" codecs="mp4a.40.2"
id="4">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate initialization="index_audio_5_0_init.mp4?
m=1566416213" media="index_audio_5_0_${Number$.mp4?m=1566416213"
presentationTimeOffset="2108797229061" startNumber="8778704" timescale="48000">
            <SegmentTimeline>
                <S d="249856" t="2108797228933"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation audioSamplingRate="48000" bandwidth="96636" codecs="mp4a.40.2"
id="5">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate initialization="index_audio_8_0_init.mp4?
m=1566416213" media="index_audio_8_0_${Number$.mp4?m=1566416213"
presentationTimeOffset="2108797229061" startNumber="8778704" timescale="48000">
            <SegmentTimeline>
                <S d="249856" t="2108797228933"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation audioSamplingRate="48000" bandwidth="64643" codecs="mp4a.40.2"
id="6">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate initialization="index_audio_26_0_init.mp4?
m=1566416213" media="index_audio_26_0_${Number$.mp4?m=1566416213"
presentationTimeOffset="2108797229061" startNumber="8778704" timescale="48000">
            <SegmentTimeline>
                <S d="249856" t="2108797228933"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
<AdaptationSet bitstreamSwitching="true" codecs="stpp" lang="eng"
mimeType="application/mp4" segmentAlignment="true" startWithSAP="1">

```

```

    <Label>eng</Label>
    <Representation bandwidth="0" id="7">
      <SegmentTemplate initialization="index_subtitles_4_0_init.mp4?
m=1566416213" media="index_subtitles_4_0_${Number$.mp4?m=1566416213"
presentationTimeOffset="3953994804010" startNumber="8778704" timescale="90000">
        <SegmentTimeline>
          <S d="468468" t="3953994804010"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
</Period>
</MPD>

```

DASH-Ortungsfunktion

Dieser Abschnitt enthält Informationen über die Speicherort-Funktion für DASH, die in AWS Elemental MediaTailor standardmäßig aktiviert ist. Lesen Sie diesen Abschnitt, wenn Sie CDN-Routingregeln (Content Delivery Network) für den Zugriff auf MediaTailor Manifeste erstellen. Informieren Sie sich zudem in diesem Abschnitt, wenn Sie die serverseitige Berichterstellung bei Playern verwenden, die keine Sticky-HTTP-Umleitungen unterstützen.

Was ist die Speicherort-Funktion?

Mit der Speicherort-Funktion können Player, die keine Sticky-HTTP-Umleitungen unterstützen, in ihren Manifest-Aktualisierungsanforderungen Sticky-Verhalten ermöglichen.

AWS Elemental MediaTailor verwendet eine sitzungslose Initialisierung und verlangt von seinen Playern ein permanentes HTTP-Umleitungsverhalten. Bei serverseitiger Berichterstattung gibt der Dienst, wenn der Spieler eine Anfrage für eine Aktualisierung des Manifests stellt MediaTailor, eine temporäre 302-Weiterleitung aus, um den Spieler an einen Endpunkt für das personalisierte Manifest weiterzuleiten. MediaTailor schließt eine Sitzungs-ID als Abfrageparameter in die Antwort ein. Beabsichtigt ist, dass der Player die URL über die gesamte Sitzung hinweg nachverfolgt, aber Player, die keine Sticky-HTTP-Umleitungen unterstützen, geben die Umleitung auf und kehren zur ursprünglichen URL zurück. Wenn ein Spieler zur ursprünglichen URL zurückkehrt, wird für jede neue Anfrage eine neue Sitzung MediaTailor erstellt, anstatt bei der ursprünglichen Sitzung zu bleiben. Dies kann zu einer Beschädigung des Manifests führen.

Die DASH-Spezifikation bietet eine Lösung für dieses Problem in der Ortungsfunktion, die in AWS Elemental MediaTailor Konfigurationen standardmäßig aktiviert ist. Wenn diese Funktion aktiviert ist, wird MediaTailor die absolute URL in das <Location> Manifest-Tag eingefügt. Player, die

keine Sticky-HTTP-Umleitungen unterstützen, können mit der in `<Location>` bereitgestellten URL Aktualisierungen am Manifest anfordern.

Muss ich die Speicherort-Funktion in meiner Konfiguration deaktivieren?

Die Speicherort-Funktion überschreibt alle CDN Routing-Regeln, die Sie für den Zugriff auf AWS Elemental MediaTailor -Manifeste einrichten. Daher müssen Sie sie möglicherweise deaktivieren. Der Speicherort-Funktion wirkt sich nicht auf das CDN-Caching von Inhalten oder Werbe-Segmenten aus.

Bestimmen Sie anhand Ihrer Situation in der folgenden Liste, ob Sie die Speicherort-Funktion für Ihre Konfiguration deaktivieren müssen und wie dabei vorzugehen ist:

- Wenn keine CDN-Routing-Regeln für den Zugriff auf AWS Elemental MediaTailor -Manifeste eingerichtet sind, lassen Sie die Speicherort-Einstellung aktiviert.
- Verwenden Sie andernfalls die folgende Regeln:
 - Wenn Sie keine serverseitige Berichterstellung verwenden oder Ihre Player alle Sticky-HTTP-Umleitungen unterstützen, deaktivieren Sie die Speicherort-Funktion. Weitere Informationen, wie dies über die Konsole möglich ist, finden Sie unter [the section called “Konfiguration erstellen”](#).
 - Andernfalls wenden Sie sich an den [AWS-Support](#).

Muss die Speicherortfunktion verwendet werden?

Sie müssen die Standortfunktion für Spieler verwenden, die keine Sticky-HTTP-Weiterleitungen unterstützen. Verwenden Sie die angegebene URL in dem `<Location>`-Tag für alle Ihre Manifest-Aktualisierungsanfragen.

Beispiel

Beispiel URLs und `<Location>` Beispiel-Tag.

- Example Beispiel: Anfängliche Anforderungs-URL

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/
dash/5ca4c1892b1f213a1247fad47b3e34c454a7d490/testLocationTag/index.mpd
```

- Example Beispiel: Umgeleiteten 302-Antwort

```
/v1/dash/5ca4c1892b1f213a1247fad47b3e34c454a7d490/testLocationTag/index.mpd?
aws.sessionId=0e5d9b45-ae97-49eb-901b-893d043e0aa6
```

- Example Beispiel: Speicherort-Tag in einem Manifest

```
<Location>https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/
dash/5ca4c1892b1f213a1247fad47b3e34c454a7d490/testLocationTag/index.mpd?
aws.sessionId=0e5d9b45-ae97-49eb-901b-893d043e0aa6</Location>
```

Sicherung der AWS Elemental MediaTailor ursprünglichen Interaktionen mit SigV4

Signature Version 4 (Sigv4) ist ein Signaturprotokoll, das zur Authentifizierung von MediaTailor Anfragen an unterstützte Quellen über HTTPS verwendet wird. Schließt bei der Sigv4-Signatur einen signierten Autorisierungsheader in die HTTPS-Ursprungsanfrage an MediaTailor Channel Assembly, Amazon S3 und AWS Elemental MediaPackage Version 2 ein. MediaTailor

Sie können SigV4 an Ihrem Ursprung verwenden, um sicherzustellen, dass Manifestanfragen nur erfüllt werden, wenn sie von einem signierten Autorisierungsheader stammen MediaTailor und einen signierten Autorisierungsheader enthalten. Auf diese Weise wird verhindert, dass nicht autorisierte MediaTailor Wiedergabekonfigurationen auf Ihre ursprünglichen Inhalte zugreifen können. Wenn der signierte Autorisierungsheader gültig ist, erfüllt Ihr Absender die Anfrage. Wenn er nicht gültig ist, schlägt die Anfrage fehl.

In den folgenden Abschnitten werden die Anforderungen für die Verwendung der MediaTailor SigV4-Signatur für unterstützte Ursprünge beschrieben.

MediaTailor Anforderungen an die Kanalmontage

Wenn Sie SigV4 verwenden, um Ihren MediaTailor Channel Assembly-Ursprung zu schützen, müssen die folgenden Anforderungen erfüllt sein, MediaTailor um auf das Manifest zugreifen zu können:

- Die ursprüngliche Basis-URL in Ihrer MediaTailor Konfiguration muss ein Channel Assembly-Channel im folgenden Format sein: `channel-assembly.mediatailor.region.amazonaws.com`
- Ihr Ursprung muss für die Verwendung von HTTPS konfiguriert sein. Wenn HTTPS am Ursprung nicht aktiviert ist, MediaTailor wird die Anfrage nicht signiert.
- Für deinen Kanal muss eine Ursprungszugriffsrichtlinie gelten, die Folgendes beinhaltet:

- Hauptzugriff für MediaTailor den Zugriff auf deinen Kanal. Gewähren Sie Zugriff auf `mediatailor.amazonaws.com`.
- IAM-Berechtigungen `mediatailor`: zum Lesen aller Manifeste der obersten Ebene, auf die in der Konfiguration verwiesen wird. `GetManifest` MediaTailor

Informationen zum Einrichten einer Richtlinie für den Kanal finden Sie unter. [Erstellen Sie mit der MediaTailor Konsole einen Kanal](#)

Example Ursprungszugriffsrichtlinie für Channel Assembly, bezogen auf das MediaTailor Konfigurationskonto

```
{
  "Effect": "Allow",
  "Principal": {"Service": "mediatailor.amazonaws.com"},
  "Action": "mediatailor:GetManifest",
  "Resource": "arn:aws:mediatailor:us-west-2:777788889999:channel/ca-origin-channel",
  "Condition": {
    "StringEquals": {"AWS:SourceAccount": "777788889999"}
  }
}
```

Example Ursprungszugriffsrichtlinie für Channel Assembly, bezogen auf die Wiedergabekonfiguration MediaTailor

```
{
  "Effect": "Allow",
  "Principal": {"Service": "mediatailor.amazonaws.com"},
  "Action": "mediatailor:GetManifest",
  "Resource": "arn:aws:mediatailor:us-west-2:777788889999:channel/ca-origin-channel",
  "Condition": {
    "StringEquals": {"AWS:SourceArn": "arn:aws:mediatailor:us-west-2:777788889999:playbackConfiguration/test"}
  }
}
```

Amazon S3 S3-Anforderungen

Wenn Sie SigV4 verwenden, um Ihren Amazon S3 S3-Ursprung zu schützen, müssen die folgenden Voraussetzungen erfüllt sein, MediaTailor um auf das Manifest zugreifen zu können:

- Die ursprüngliche Basis-URL in Ihrer MediaTailor Konfiguration muss ein S3-Bucket im folgenden Format sein: `s3.region.amazonaws.com`
- Ihr Ursprung muss für die Verwendung von HTTPS konfiguriert sein. Wenn HTTPS am Ursprung nicht aktiviert ist, MediaTailor wird die Anfrage nicht signiert.
- Für deinen Kanal muss eine Ursprungszugriffsrichtlinie gelten, die Folgendes beinhaltet:
 - Hauptzugriff für den MediaTailor Zugriff auf deinen Bucket. Gewähren Sie Zugriff auf `mediatailor.amazonaws.com`.

Informationen zur Konfiguration des Zugriffs in IAM finden Sie unter [Zugriffsverwaltung](#) im AWS Identity and Access Management-Benutzerhandbuch.

- IAM-Berechtigungen `s3:GetObject` zum Lesen aller Manifeste der obersten Ebene, auf die in der Konfiguration verwiesen wird. MediaTailor

Allgemeine Informationen zu SigV4 für Amazon S3 finden Sie im Thema [Authenticating Requests \(AWS Signature Version 4\)](#) in der Amazon S3 S3-API-Referenz.

Example Ursprungszugriffsrichtlinie für Amazon S3, bezogen auf das Konto MediaTailor

```
{
  "Effect": "Allow",
  "Principal": {"Service": "mediatailor.amazonaws.com"},
  "Action": "s3:GetObject",
  "Resource": "arn:aws:s3:::mybucket/*",
  "Condition": {
    "StringEquals": {"AWS:SourceAccount": "111122223333"}
  }
}
```

Example Origin-Zugriffsrichtlinie für Amazon S3, bezogen auf die Wiedergabekonfiguration MediaTailor

```
{
  "Effect": "Allow",
  "Principal": {"Service": "mediatailor.amazonaws.com"},
  "Action": "s3:GetObject",
  "Resource": "arn:aws:s3:::mybucket/*",
  "Condition": {
    "StringEquals": {"AWS:SourceArn": "arn:aws:mediatailor:us-west-2:111122223333:playbackConfiguration/test"}
  }
}
```

```
}
}
```

MediaPackage Anforderungen

Wenn Sie SigV4 verwenden, um Ihren MediaPackage v2-Ursprung zu schützen, müssen die folgenden Voraussetzungen erfüllt sein, MediaTailor um auf das Manifest zugreifen zu können:

- Die ursprüngliche Basis-URL in Ihrer MediaTailor Konfiguration muss ein MediaPackage v2-Endpunkt im folgenden Format sein: `mediapackagev2.region.amazonaws.com`
- Ihr Ursprung muss für die Verwendung von HTTPS konfiguriert sein. Wenn HTTPS am Ursprung nicht aktiviert ist, MediaTailor wird die Anfrage nicht signiert.
- Für deinen Kanal muss eine Ursprungszugriffsrichtlinie gelten, die Folgendes beinhaltet:
 - Hauptzugriff für den MediaTailor Zugriff auf deinen Endpunkt. Gewähren Sie Zugriff auf `mediatailor.amazonaws.com`.
 - IAM-Berechtigungen `mediapackagev2`: zum Lesen aller Manifeste der obersten Ebene, auf die in der Konfiguration verwiesen wird. `GetObject` MediaTailor

Allgemeine Informationen zu SigV4 für MediaPackage v2 finden Sie im Thema [Authenticating Requests \(AWS Signature Version 4\)](#) in der MediaPackage v2-API-Referenz.

Example Origin-Zugriffsrichtlinie für MediaPackage Version 2, bezogen auf das Konto MediaTailor

```
{
  "Effect": "Allow",
  "Principal": {"Service": "mediatailor.amazonaws.com"},
  "Action": "mediapackagev2:GetObject",
  "Resource": "arn:aws:mediapackagev2:us-west-2:444455556666:channelGroup/emp-origin-channel-group/channel/emp-origin-channel/originEndpoint/emp-origin-endpoint",
  "Condition": {
    "StringEquals": {"AWS:SourceAccount": "444455556666"}
  }
}
```

Example Origin-Zugriffsrichtlinie für MediaPackage Version 2, bezogen auf die Wiedergabekonfiguration MediaTailor

```
{
```

```
"Effect": "Allow",
"Principal": {"Service": "mediatailor.amazonaws.com"},
"Action": "mediapackagev2:GetObject",
"Resource": "arn:aws:mediapackagev2:us-west-2:444455556666:channelGroup/emp-origin-
channel-group/channel/emp-origin-channel/originEndpoint/emp-origin-endpoint",
"Condition": {
  "StringEquals": {"AWS:SourceArn": "arn:aws:mediatailor:us-
west-2:444455556666:playbackConfiguration/test""}
}
```

Integration AWS Elemental MediaTailor mit Google Ad Manager

Durch die Integration MediaTailor mit [Google Ad Manager](#) (Ad Manager) erhalten Sie programmatischen Zugriff auf einen auktionsgesteuerten Online-Marktplatz, auf dem Anzeigenimpressionen in Echtzeit gekauft und verkauft werden können. Sie müssen ein Konto bei Ad Manager eingerichtet haben. Anschließend können Sie die Integration mit Ad Manager auf folgende Weise durchführen:

- Eine serverseitige Integration mit einem SSL-Zertifikat.
- Eine clientseitige Player-Integration unter Verwendung des Programmatic Access Libraries (PAL) - SDK. Diese Integration ist erforderlich, wenn Sie den Transaktionstyp „Offene Auktion“ verwenden möchten.

Die Ad Manager-Unterstützung für programmatische Transaktionstypen hängt von der Art der Integration ab, die Sie verwenden. Eine Liste der verfügbaren Optionen finden Sie unter [Transaktionstypen](#) oder wenden Sie sich an Ihr Google-Kontoteam.

In den folgenden Abschnitten werden diese Integrationen detailliert beschrieben.

Themen

- [Serverseitige AWS Elemental MediaTailor Integration mit Google Ad Manager](#)
- [Kundenseitige AWS Elemental MediaTailor Integration mit Google Ad Manager](#)

Serverseitige AWS Elemental MediaTailor Integration mit Google Ad Manager

Serverseitige Anzeigenanfragen an Google Ad Manager (Ad Manager) müssen das SSL-Zertifikat enthalten, für das Ad Manager ausgestellt hat, um programmatische Transaktionen MediaTailor zu autorisieren.

Um serverseitige Anzeigenanfragen an Ad Manager zu stellen

1. [Reichen Sie ein AWS Support-Ticket](#) ein, um die Aktivierung von SSL-Zertifikaten anzufordern. Nehmen Sie die folgenden Informationen in das Support-Ticket auf:
 - AWS Region
 - AWS Konto-ID
 - MediaTailor Name der Wiedergabekonfiguration

Wenn Sie keine SSL-Zertifikate aktivieren, reagiert Ad Manager auf MediaTailor Anzeigenanfragen mit HTTP 401-Fehlercodes im Ereignistyp `ERROR_ADS_INVALID_RESPONSE` ADS-Interaktionsprotokoll.

2. Nachdem die SSL-Zertifikate aktiviert wurden, aktualisieren Sie die URL und die Parameter für Ihre ADS- und Preroll-ADS in der Wiedergabekonfiguration. Informationen zum Aktualisieren oder Erstellen einer Wiedergabekonfiguration finden Sie unter [Mit Wiedergabekonfigurationen arbeiten AWS Elemental MediaTailor](#).

Offizielle Hinweise zu URL-Parametern für VAST-Anzeigenanfragen für Ad Manager finden Sie im [Leitfaden zur serverseitigen Implementierung](#) von Ad Manager. Die Aktualisierung beinhaltet die folgenden Änderungen:

- Ändern Sie die Basis-URL von `pubads.g.doubleclick.net` zu `zuserside.doubleclick.net`.
- Fügen Sie den `ssss=mediatailor`-Parameter hinzu. Dies weist darauf hin, dass MediaTailor es sich um die serverseitige Stitching-Quelle handelt.
- Entfernen Sie den Parameter `IP`. MediaTailor übergibt automatisch die IP-Adresse des Endbenutzers mithilfe des `X-Forwarded-For` Headers.
- Entfernen Sie den `ss_req=1` Parameter.

Aktualisierte und vollständige VAST-URL-Anleitungen finden Sie im [Leitfaden zur serverseitigen Implementierung](#) oder wenden Sie sich an Ihr Google-Kontoteam.

Kundenseitige AWS Elemental MediaTailor Integration mit Google Ad Manager

Für die Verwendung der Google Ad Manager Programmatic Access Libraries (PAL) ist eine MediaTailor clientseitige Integration erforderlich. SDKs Diese Integration ist erforderlich, wenn Sie den Transaktionstyp „Offene Auktion“ von Ad Manager verwenden möchten.

Die PAL SDKs enthalten Informationen über den Inhalt, das Gerät und die Benutzerdaten für eine Wiedergabesitzung. Über das PAL-SDK können Sie diese Informationen an Google Ad Manager weitergeben, der dann besser bestimmen kann, welche zielgerichteten Anzeigen geschaltet werden sollen. SDKs sind für Android HTML5, iOS und Cast verfügbar. Informationen zur Verwendung von PAL SDKs finden Sie unter [Google Ad Manager PAL SDK](#).

So erstellen Sie eine clientseitige Integration mit Ad Manager

1. Verwenden Sie das PAL-SDK, um eine Nonce zu generieren.

Die Nonce ist eine verschlüsselte Zeichenfolge, die PAL für Stream-Anfragen generiert. Jede Anfrage muss eine eindeutige Nonce haben. Informationen zur Einrichtung einer Nonce erhalten Sie, wenn Sie Ihr SDK aus dem [Google Ad Manager PAL](#) SDK auswählen.

2. Verwenden Sie den `givn` Parameter in Ihrer ADS-Anfrage, um den Nonce-Wert zu übergeben. Aktualisieren Sie dazu Ihre ADS-URL so, dass sie Folgendes enthält:
&givn=[*player_params.givn*] Detaillierte Anweisungen finden Sie unter [Aktivierung der clientseitigen Nachverfolgung](#).

Datazoom-Player SDKs

MediaTailor hat sich mit Datazoom zusammengetan, um einen kostenlosen Player zur Verfügung zu stellen, mit SDKs der Integrationen, SDKs wie sie beispielsweise im Ad Manager PAL angeboten werden, vereinfacht werden können. Informationen zum Datazoom und zur Partnerschaft finden Sie unter MediaTailor [Kostenloser Datazoom-Player SDKs](#)

Um auf den Datazoom Player zuzugreifen SDKs, verwenden Sie die Kontaktinformationen auf der [Datazoom](#) with AWS-Website.

Verwendung eines CDN zur Optimierung der Anzeigenpersonalisierung und Inhaltsbereitstellung

Wir empfehlen Ihnen dringend, ein Content Distribution Network (CDN) wie Amazon CloudFront zu verwenden, um die Effizienz des Workflows zur Anzeigenpersonalisierung und Kanalzusammenstellung zwischen AWS Elemental MediaTailor und Ihren Nutzern zu verbessern. Zu den Vorteilen eines CDN gehören Zwischenspeicherung von Inhalten und Werbung, konsistente Domännennamen in personalisierten Manifesten und CDN-DNS-Auflösung.

Wenn Sie im AWS Elemental MediaTailor Workflow ein CDN verwenden, sieht der Anforderungs- und Antwortablauf wie folgt aus:

1. Der Spieler fordert vom CDN ein Manifest an, dessen Ursprung das Manifest ist. MediaTailor Das CDN leitet die Anfrage weiter an. MediaTailor
2. MediaTailor personalisiert das Manifest und ersetzt die URL-Präfixe für Inhalt und Anzeigensegment durch CDN-Domännennamen. MediaTailor sendet das personalisierte Manifest als Antwort an das CDN, das es an den anfragenden Spieler weiterleitet.
3. Der Spieler fordert Segmente aus den an URLs , die im Manifest enthalten sind.
4. Das CDN übersetzt das Segment. URLs Es leitet Anfragen zu Inhaltssegmenten an den Ursprungsserver und Anzeigenanfragen an die CloudFront Amazon-Distribution weiter, in der transkodierte Anzeigen MediaTailor gespeichert werden.
5. Der Ursprungsserver MediaTailor antwortet mit den angeforderten Segmenten, und die Wiedergabe beginnt.

In den folgenden Abschnitten werden die Konfiguration AWS Elemental MediaTailor und das CDN zur Durchführung dieses Ablaufs beschrieben.

Themen

- [Ein CDN integrieren](#)
- [Wie AWS Elemental MediaTailor geht Base for DASH URLs um](#)
- [Bewährte CDN-Praktiken mit AWS Elemental MediaTailor](#)

Ein CDN integrieren

Die folgenden Schritte zeigen, wie Sie die AWS Elemental MediaTailor Integration in Ihr Content Distribution Network (CDN) durchführen können. Je nach dem von Ihnen verwendeten CDN unterscheiden sich einige Begriffe möglicherweise von den in diesen Schritten genannten.

Schritt 1: (CDN) Routing-Verhalten erstellen

Erstellen Sie im CDN Verhaltensweisen und Regeln, die Wiedergabeanforderungen an MediaTailor weiterleiten. Verwenden Sie die folgenden Regeln für alle Segmentanforderungen (Inhalt, normale Ad-Avails und Pre-Roll-Ad-Avails):

- Erstellen Sie eine Verhaltensweise, die Inhaltssegment-Anforderungen an den Ursprungs-Server weiterleitet. Basieren Sie diese auf einer Regel, die eine Formulierung zur Unterscheidung von Inhaltssegment-Anforderungen von Werbesegment-Anforderungen verwendet.

Beispiel: Das CDN könnte HLS-Player-Anforderungen an `https://CDN_Hostname/subdir/content.ts`, basierend auf dem Schlüsselwort `subdir` in der Anforderung, zum Pfad `http://origin.com/contentpath/subdir/content.ts` des Ursprungsservers weiterleiten.

Beispiel: Das CDN könnte DASH-Player-Anforderungen an `https://CDN_Hostname/subdir/content.mp4`, basierend auf dem Schlüsselwort `subdir` in der Anforderung, zum Pfad `http://origin.com/contentpath/subdir/content.mp4` des Ursprungs-Servers weiterleiten.

- (Optional) Erstellen Sie ein Verhalten, das Anzeigensegmentanfragen an den internen CloudFront Amazon-Vertrieb weiterleitet, in dem transkodierte Anzeigen AWS Elemental MediaTailor gespeichert werden. Basieren Sie diese auf einer Regel, die eine Formulierung zur Unterscheidung von Werbesegment-Anforderungen von Inhaltssegment-Anforderungen enthält. Dieser Schritt ist optional, da er eine Standardkonfiguration AWS Elemental MediaTailor bietet.

AWS Elemental MediaTailor verwendet die folgenden CloudFront Standard-Amazon-Distributionen zum Speichern von Anzeigen:

Example Routing von Anzeigensegmenten

Pattern: `https://segments.mediatailor.<region>.amazonaws.com`

Beispiel: `https://segments.mediatailor.eu-west-1.amazonaws.com`

Schritt 2: (AWS Elemental MediaTailor) Erstellen Sie eine Konfiguration mit CDN-Mapping

Erstellen Sie eine AWS Elemental MediaTailor Konfiguration, die die Domänen des CDN-Routing-Verhaltens dem Ursprungsserver und dem Anzeigenspeicherort zuordnet. Geben Sie die Domännennamen wie folgt in die Konfiguration ein:

- Geben Sie für CDN content segment prefix (CDN-Inhaltssegment-Präfix) die CDN-Domäne aus der Verhaltensweise ein, die Sie zur Weiterleitung der Inhaltsanforderungen an den Ursprungs-Server erstellt haben. MediaTailor Ersetzt im Manifest das URL-Präfix für das Inhaltssegment durch die CDN-Domain.

Erwägen wir z. B. die folgenden Einstellungen.

- Video content source (Quelle der Videoinhalte) in der MediaTailor-Konfiguration ist `http://origin.com/contentpath/`
- CDN content segment prefix (CDN-Inhaltssegment-Präfix) is `https://CDN_Hostname/`

Für HLS gilt: Wenn der Pfad der vollständigen Inhaltsdatei lautet `http://origin.com/contentpath/subdir/content.ts`, ist das Inhaltssegment im Manifest, das von MediaTailor bereitgestellt wird. `https://CDN_Hostname/subdir/content.ts`

Für DASH gilt: Wenn der Pfad der vollständigen Inhaltsdatei lautet `http://origin.com/contentpath/subdir/content.mp4`, ist das Inhaltssegment im Manifest, das von MediaTailor bereitgestellt wird. `https://CDN_Hostname/subdir/content.mp4`.

- Geben Sie für CDN ad segment prefix (CDN-Werbesegment-Präfix) den Namen der CDN-Verhaltensweise ein, die Sie zur Weiterleitung von Werbeanforderungen über Ihr CDN erstellt haben. MediaTailor Ersetzt im Manifest die CloudFront Amazon-Distribution durch den Verhaltensnamen.

Schritt 3: (CDN) Richten Sie das CDN für Manifest- und Berichtsabfragen ein

Durch Verwendung eines CDN für Manifest- und Berichterstellungsabfragen wird Ihnen in Ihrem Workflow mehr Funktionalität geboten.

Wenn Sie bei Manifesten noch vor der Manifestspezifikation auf ein CDN verweisen, können Sie CDN-Funktionen wie Geofencing nutzen und außerdem alles von Ihrem eigenen Domännennamen aus bedienen. Speichern Sie die Manifeste für diesen Pfad nicht zwischen, da sie alle personalisiert sind. Manifest-Spezifikationen befinden sich unter `/v1/master` für HLS-Master-

Manifestanforderungen, unter `/v1/manifest` für HLS-Medien-Manifestanforderungen und unter `/v1/dash` für DASH Manifestanforderungen.

Stellen Sie sicher, dass Ihr CDN alle Abfrageparameter an weiterleitet. AWS Elemental MediaTailor MediaTailor stützt sich auf die Abfrageparameter, um Ihre VAST-Anfragen für personalisierte Anzeigen zu erfüllen.

Bei serverseitigen Berichten trägt der Verweis auf ein CDN vor Anfragen `/v1/segment` in Anzeigensegmenten dazu bei, dass keine doppelten AWS Elemental MediaTailor Anzeigen-Tracking-Beacons gesendet werden. Wenn ein Player eine Anforderung für eine `/v1/segment`-Werbung stellt, gibt MediaTailor eine 301-Umleitung an das tatsächliche `*.ts`-Segment aus. Wenn MediaTailor diese `/v1/segment`-Anforderung erkennt, gibt es einen Beacon-Aufruf zur Nachverfolgung des Prozentsatzes der angesehenen Werbung aus. Wenn derselbe Player `/v1/segment` in einer Sitzung mehrere Anfragen für dasselbe Objekt stellt und Ihr Ad Decision Server (ADS) Anfragen nicht deduplizieren kann, MediaTailor gibt er mehrere Anfragen für dasselbe Beacon aus. Durch Zwischenspeichern dieser 301-Antworten mit einem CDN wird MediaTailor an doppelten Beacon-Aufrufen für wiederholte Anforderungen gehindert. Für diesen Pfad können Sie einen hohen oder einen Standard-Cache verwenden, da die Cache-Schlüssel für diese Segmente eindeutig sind.

Um diese Vorteile zu nutzen, erstellen Sie im CDN Verhaltensweisen, die Anfragen an den Konfigurationsendpunkt weiterleiten. AWS Elemental MediaTailor Unterstellen Sie die von Ihnen erstellten Verhaltensweisen Regeln, die zwischen Anforderungen für Master-HLS-Manifeste, HLS-Manifeste, DASH-Manifeste und Berichterstellung unterscheiden.

Für Anforderungen gelten die folgenden Formate:

- HLS-Master-Manifestformat

```
https://<playback-endpoint>/v1/master/<hashed-account-id>/<origin-id>/<master>.m3u8
```

Beispiel

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/master/  
a1bc06b59e9a570b3b6b886a763d15814a86f0bb/Demo/assetId.m3u8
```

- HLS-Manifest-Format

```
https://<playback-endpoint>/v1/manifest/<hashed-account-id>/<session-id>/  
<manifestNumber>.m3u8
```

Beispiel

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/manifest/
a1bc06b59e9a570b3b6b886a763d15814a86f0bb/c240ea66-9b07-4770-8ef9-7d16d916b407/0.m3u8
```

- DASH-Manifestformat

```
https://<playback-endpoint>/v1/dash/<hashed-account-id>/<origin-id>/<assetName>.mpd
```

Beispiel

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/dash/
a1bc06b59e9a570b3b6b886a763d15814a86f0bb/Demo/0.mpd
```

- Format der Werbebericht-Anforderung für die serverseitige Berichterstellung

```
https://<playback-endpoint>/v1/segment/<origin-id>/<session-id>/<manifestNumber>/
<HLSSequenceNum>
```

Beispiel

```
https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/segment/
Demo/240ea66-9b07-4770-8ef9-7d16d916b407/0/440384
```

Erstellen Sie im CDN eine Verhaltensweise, mit der Manifestanforderungen an den AWS Elemental MediaTailor -Konfigurationsendpunkt weitergeleitet werden. Basieren Sie die Verhaltensweise auf einer Regel, die eine Formulierung zur Unterscheidung der Manifest-Anforderung von Segment-Anforderungen enthält.

Example Routing

- Spieleranfragen an `https://CDN_Hostname/some/path/asset.m3u8` werden auf der `https://mediatailor.us-west-2.amazonaws.com/v1/session/configuration/endpoint` Grundlage des Schlüsselworts `*.m3u8` in der Anfrage an den AWS Elemental MediaTailor Pfad weitergeleitet.
- Spieleranfragen `https://CDN_Hostname/some/path/asset.mpd` werden auf der `https://mediatailor.us-west-2.amazonaws.com/v1/dash/configuration/endpoint`

Grundlage des Schlüsselworts * .mpd in der Anfrage an den AWS Elemental MediaTailor Pfad weitergeleitet.

Wie AWS Elemental MediaTailor geht Base for DASH URLs um

Bei der serverseitigen Werbeeinblendung stammen die Inhaltssegmente und die Werbesegmente von verschiedenen Speicherorten. AWS Elemental MediaTailor Verwaltet in Ihren DASH-Manifesten die URL-Einstellungen auf der Grundlage Ihrer CDN-Konfiguration (Content Distribution Network) und den URLs Angaben im Manifest. MediaTailor verwendet die Regeln in der folgenden Liste, um die BaseURL Einstellungen in Ihren DASH-Manifesten für Ihre Inhaltssegmente und Anzeigensegmente zu verwalten.

AWS Elemental MediaTailor Verhalten für Inhaltssegmente:

- Wenn Sie in Ihrer Konfiguration ein CDN-Inhaltssegment-Präfix angeben, MediaTailor stellen Sie sicher BaseURL, dass genau eines mit Ihrem angegebenen Präfix auf der MPD Ebene definiert ist.
- Wenn Sie kein CDN content segment prefix (CDN-Inhaltssegment-Präfix) angeben, dann verwendet MediaTailor wie folgt das Ursprungs-Vorlagen-Manifest:
 - Wenn das Ursprungs-Vorlagen-Manifest eine oder mehrere BaseURL-Einstellungen auf der MPD-Ebene enthält, behält MediaTailor sie unverändert bei.
 - Wenn das Ursprungs-Vorlagen-Manifest keine BaseURL-Einstellung auf MPD-Ebene enthält, dann fügt MediaTailor eine hinzu, die auf der Ursprungs-MPD-URL basiert.

Geht für Anzeigensegmente AWS Elemental MediaTailor wie folgt vor:

- Wenn Sie in Ihrer Konfiguration ein CDN-Anzeigensegmentpräfix angeben, stellen Sie MediaTailor sicher, dass jeder Anzeigenzeitraum genau eine BaseURL Einstellung hat, die mit dem konfigurierten Präfix aufgefüllt wird.
- Wenn Sie kein CDN-Anzeigensegmentpräfix angeben, MediaTailor fügt es jedem Anzeigenzeitraum genau eine BaseURL Einstellung hinzu, die auf den Anzeigeninhaltsserver verweist, der von MediaTailor für die Bereitstellung von Anzeigensegmenten eingerichtet wurde.

Bewährte CDN-Praktiken mit AWS Elemental MediaTailor

Es wird dringend empfohlen, dass Sie ein Content Distribution Network (CDN) verwenden, um Inhalte und Werbesegmente zwischenspeichern. Personalisierte Manifestantworten dürfen jedoch

nicht zwischengespeichert oder zwischen Viewern geteilt werden. Verwenden Sie die folgenden Einstellungen für Manifestdatenverkehr in Ihrem CDN, um den Service optimal nutzen zu können:

- Legen Sie alle Time to Live (TTL)-Einstellungen auf **0** fest. Dies umfasst die maximale, minimale und standardmäßige TTL.
- Leiten Sie alle Abfragezeichenfolgen weiter an MediaTailor. Auf diese Weise können alle Werbevariablen an den Ad Decision Server (ADS) übergeben werden, um zu bestimmen, welche Werbung in dieser Wiedergabesitzung verwendet werden soll.
- Leitet den **User-Agent** Header weiter an MediaTailor. Der ADS muss oft wissen, welcher Benutzer-Agent den Inhalt anfordert. Wenn Sie den User-Agent-Header nicht weiterleiten, ist der von MediaTailor empfangene Wert der Benutzer-Agent Ihres CDN.

Anpassen des Verhaltens von Werbeunterbrechungen mit Anzeigenunterdrückung

Wenn Sie eine Konfiguration in erstellen AWS Elemental MediaTailor, können Sie optionale Konfigurationseinstellungen für Werbeunterbrechungen angeben, die das Verhalten von Werbeunterbrechungen steuern, einschließlich der Möglichkeit, die Unterdrückung von Werbeunterbrechungen zu konfigurieren. Auf diese Weise können Sie die Werbeblock-Erlebnisse für Ihre Videoinhalte an Ihre spezifischen Anforderungen anpassen.

Themen

- [Die Unterdrückung von Werbeunterbrechungen konfigurieren](#)

Die Unterdrückung von Werbeunterbrechungen konfigurieren

Note

Die Unterdrückung von Werbung ist nur für Live-Workflows verfügbar.

Sie können so konfigurieren MediaTailor , dass die Personalisierung von Werbeunterbrechungen für Live-Inhalte übersprungen wird. Dies wird als Werbeunterdrückung oder auch Werbeplatzunterdrückung bezeichnet. In diesem Thema erfahren Sie, wie das geht, und es wird auch erklärt, wie die Konfiguration der Anzeigenunterdrückung funktioniert.

Die Unterdrückung von Anzeigen kann für die folgenden Anwendungsfälle verwendet werden:

- Betrachtungszeitraum für große Manifeste – Wenn ein Betrachter die Wiedergabe an der Live-Grenze eines Manifests startet, der Betrachtungszeitraum aber groß ist, empfiehlt es sich, Werbung erst ab dem Zeitpunkt einzufügen, an dem der Betrachter mit der Wiedergabe begonnen hat. Oder fügen Sie Werbeanzeigen für einen Teil des gesamten Betrachtungszeitraum im Manifest ein. Sie können die Anzeigenunterdrückung so konfigurieren, dass Werbeunterbrechungen innerhalb oder innerhalb eines bestimmten Zeitraums hinter dem Live-Edge MediaTailor personalisiert werden.
- Teilnahme mitten in der Pause – Wenn der Betrachter mit der Ansicht eines Live-Videostream mitten in einer Werbepause beginnt, ist die Wahrscheinlichkeit groß, dass dieser Benutzer den Kanal wechselt und sich die Werbung nicht ansieht. Mit der Anzeigenunterdrückungsfunktion können Sie die Personalisierung von Werbeunterbrechungen überspringen, wenn die Werbeunterbrechung begann, bevor der Zuschauer dem Stream beigetreten ist.

Konfigurieren von Anzeigen

Um die Anzeigenunterdrückungsfunktion zu verwenden, konfigurieren Sie auf folgende Weise einen Verfügbarkeitsunterdrückungsmodus, einen Verfügbarkeitsunterdrückungswert und eine Richtlinie zum Ausfüllen der Unterdrückung:

- In der Konsole MediaTailor
- Mit dem AWS Command Line Interface (AWS CLI)
- Verwenden Sie die MediaTailor API oder als Parameter in der Anfrage für die Wiedergabe-Sitzung Ihres Clients

Hinweise zur Konfiguration mit Parametern finden Sie unter [Konfigurieren von Werbeunterdrückungsparametern – Wiedergabe-Sitzungsanforderung](#).

Konfigurationsparameter für die Anzeigenunterdrückung

Sie können wählen, ob Sie die Unterdrückung von Anzeigen aktivieren oder deaktivieren möchten. Wenn Sie die Werbeunterdrückung aktivieren, geben Sie an, ob diese Unterdrückung nach dem Rand der Live-Wiedergabe oder vor dem Rand der Live-Wiedergabe eines Live-Streams erfolgt. In beiden Fällen geben Sie auch eine Zeit an, die sich auf den Live-Rand bezieht und in der Werbung MediaTailor nicht personalisiert wird. Wenn Sie die Unterdrückung von Inhalten aktivieren, können Sie eine Richtlinie zur Unterdrückung von Inhalten angeben, die dafür MediaTailor verwendet wird,

dass Werbeunterbrechungen nur teilweise aufgefüllt werden, wenn eine Sitzung mitten in der Pause beginnt.

Im Folgenden sind die Konfigurationsparameter für die Werbeunterdrückung aufgeführt:

- **Avail suppression mode (Werbeplatz-Unterdrückungsmodus)** – Legt den Werbeunterdrückungsmodus fest. Standardmäßig ist die Werbeunterdrückung deaktiviert. Zulässige Werte: OFF, oder. BEHIND_LIVE_EDGE AFTER_LIVE_EDGE
 - **OFF:** Es gibt keine Werbeunterbrechung und alle Werbeunterbrechungen werden MediaTailor personalisiert.
 - **BEHIND_LIVE_EDGE:** personalisiert MediaTailor keine Werbeunterbrechungen, die vor dem Live-Edge beginnen, abzüglich des Werts für die Avail-Unterdrückung. Dies wirkt sich auf die gesamte Werbeunterbrechung aus, nicht nur auf einzelne Werbeanzeigen.
 - **AFTER_LIVE_EDGE:** Personalisiert MediaTailor keine Werbeunterbrechungen, die innerhalb des Live-Randes liegen, plus den Avail-Unterdrückungswert. Dieser Wert kann so konfiguriert werden, dass entweder ganze Werbeunterbrechungen oder nur ein teilweises Ausfüllen von Werbeunterbrechungen möglich ist.
- **Avail suppression value (Werbeplatz-Unterdrückungswert)** – Ein Zeitpunkt relativ zur Live-Grenze in einem Livestream. Akzeptierter Wert: Ein Zeitwert in HH:MM:SS.
- **Richtlinie zur Unterdrückung von Verfügbarkeiten** — Definiert die Richtlinie, die für den Unterdrückungsmodus Avail MediaTailor gilt. Akzeptierte Werte: PARTIAL_AVAIL, FULL_AVAIL_ONLY.
 - **BEHIND_LIVE_EDGE** Der Modus verwendet immer die FULL_AVAIL_ONLY Unterdrückungsrichtlinie.
 - **AFTER_LIVE_EDGE** Der Modus kann verwendet werden, um das Auffüllen von PARTIAL_AVAIL Werbeunterbrechungen aufzurufen, wenn eine Sitzung mitten in der Pause beginnt.

Beispiele für Einstellungen zur Anzeigenunterdrückung

Durch die Art und Weise, wie die [Konfigurationsparameter für die Anzeigenunterdrückung](#) miteinander interagieren, können Sie verschiedene Methoden für die Anzeigenunterdrückung angeben und das Ausfüllen vor, am oder nach dem Live-Rand des Livestreams nutzen. Dieser Abschnitt enthält Beispiele, die zeigen, wie diese Interaktionen sind. Verwenden Sie diese Beispiele, um die Konfigurationsparameter für Ihre spezielle Situation einzurichten.

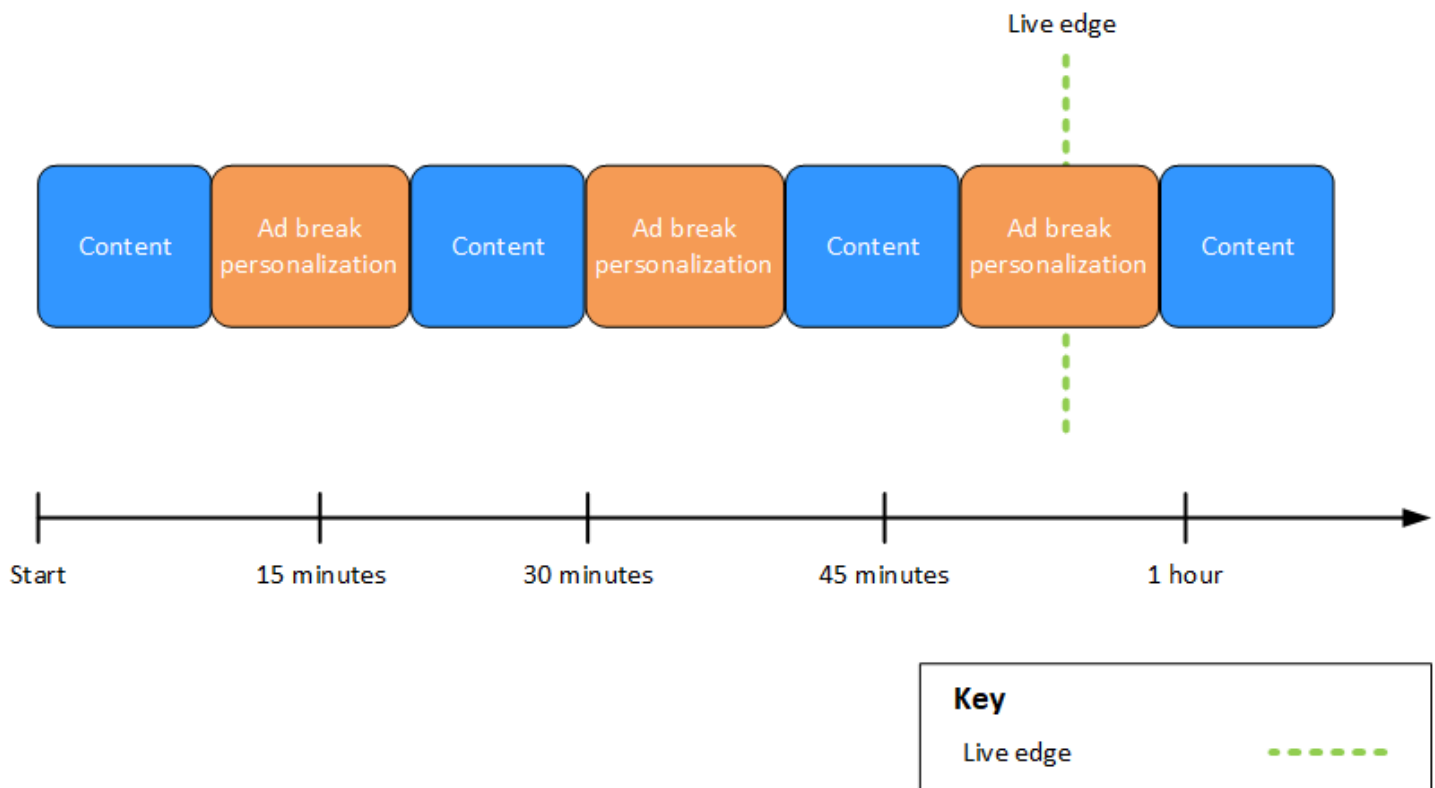
Im Folgenden finden Sie Beispiele für Einstellungen zur Anzeigenunterdrückung:

Example 1: Keine Werbeunterdrückung

Wenn der Avail-Unterdrückungsmodus aktiviert ist `OFF`, gibt es keine Werbeunterbrechung und alle Werbeunterbrechungen werden MediaTailor personalisiert.

In der folgenden Abbildung sind verschiedene Blöcke horizontal entlang einer Zeitleiste angeordnet, die von links nach rechts verläuft. Jeder Block steht für einen Zeitraum, in dem der Inhalt des Livestreams oder einer personalisierten Werbepause wiedergegeben wird. Eine gepunktete Linie steht für den aktuellen Live-Rand des Live-Streams. Vor dem Live-Edge finden zwei Werbeunterbrechungen statt, und am Live-Edge wird gerade eine weitere Werbeunterbrechung ausgeführt. Wie in der Abbildung dargestellt, werden bei aktiviertem Verfügbarkeitsunterdrückungsmodus alle Werbeunterbrechungen `OFF`, die vor dem Live-Edge stattfinden, auf der Timeline MediaTailor personalisiert. MediaTailor personalisiert auch die Werbeunterbrechung, die gerade am Live-Edge läuft.

Avail suppression mode (default): `OFF`



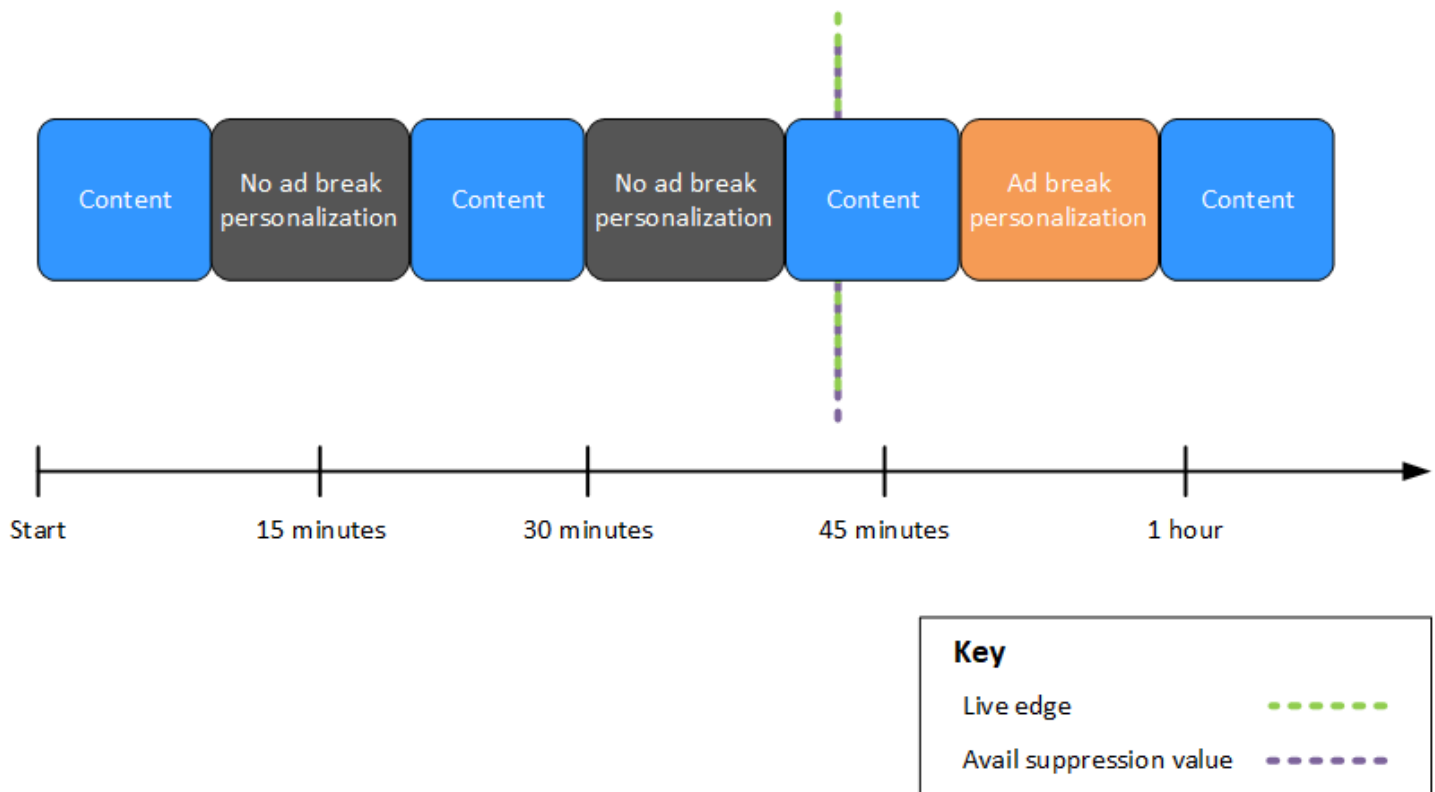
Example 2: BEHIND_LIVE_EDGE Werbeunterdrückung, deren Wert mit Live Edge synchron ist

Wenn der Verfügungsunterdrückungsmodus auf **BEHIND_LIVE_EDGE** und der Wert für die Verfügbarkeitsunterdrückung auf eingestellt ist **00:00:00**, ist der Verfügbarkeitsunterdrückungswert mit dem Live-Edge synchronisiert. MediaTailor personalisiert keine Werbeunterbrechungen, die am oder vor dem Live-Edge beginnen.

In der folgenden Abbildung sind verschiedene Blöcke horizontal entlang einer Zeitleiste angeordnet, die von links nach rechts verläuft. Jeder Block steht für einen Zeitraum, in dem der Inhalt eines Live-Streams, einer personalisierten Werbepause oder einer nicht personalisierten Werbepause abgespielt wird. Eine gepunktete Linie steht für den aktuellen Live-Rand des Live-Streams. Eine weitere gepunktete Linie, die den auf eingestellten Wert für die Funktionsunterdrückung darstellt **00:00:00**, überlappt die gepunktete Linie für den Live-Rand. Zwei Werbeunterbrechungen treten vor der Live-Kante auf, und eine weitere Werbeunterbrechung erfolgt nach der Live-Kante. Wie in der Abbildung dargestellt, werden Werbeunterbrechungen, die vor dem Live-Edge auftreten **BEHIND_LIVE_EDGE**, auf der Timeline MediaTailor nicht personalisiert, wenn der Avail-Unterdrückungsmodus auf eingestellt ist und der Avail-Unterdrückungswert **00:00:00** so eingestellt ist, dass er mit dem Live-Edge synchronisiert ist. MediaTailor personalisiert die Werbeunterbrechung, die nach dem Live-Edge erfolgt.

Avail suppression mode: **BEHIND_LIVE_EDGE**

Avail suppression value: **00:00:00**



Example 3: **BEHIND_LIVE_EDGE** Werbeunterdrückung mit einem Wert hinter dem Live-Edge

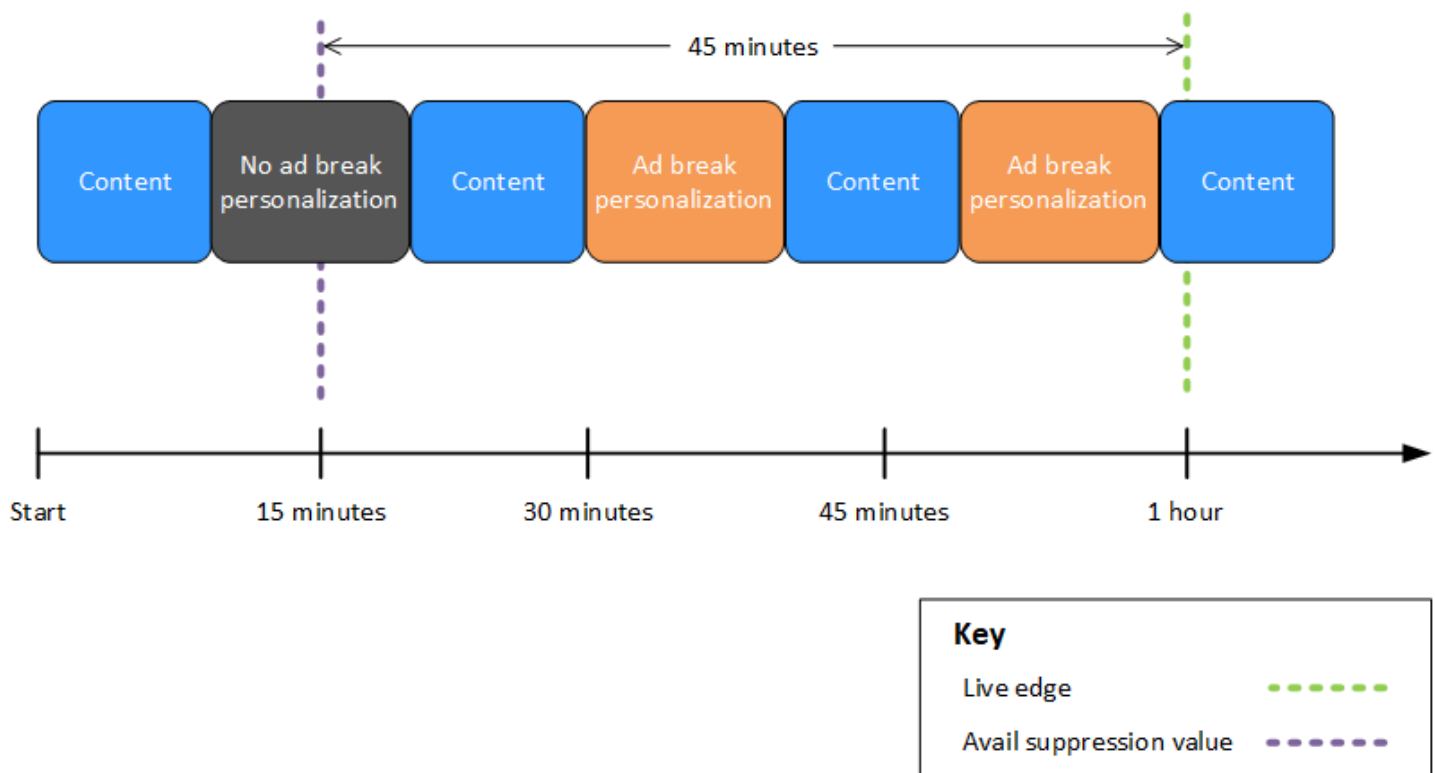
Wenn der Verfügbarkeitsunterdrückungsmodus aktiviert ist **BEHIND_LIVE_EDGE**, werden MediaTailor keine Werbeunterbrechungen an oder vor diesem Zeitpunkt personalisiert. MediaTailor personalisiert in diesem Beispiel Werbeunterbrechungen, die bis zu 45 Minuten hinter dem Live-Edge beginnen. MediaTailor personalisiert keine Werbeunterbrechungen, die mehr als 45 Minuten hinter dem Live-Edge beginnen.

In der folgenden Abbildung sind verschiedene Blöcke horizontal entlang einer Zeitleiste angeordnet, die von links nach rechts verläuft. Jeder Block steht für einen Zeitraum, in dem der Inhalt eines Live-Streams, einer personalisierten Werbepause oder einer nicht personalisierten Werbepause abgespielt wird. Eine gepunktete Linie steht für den aktuellen Live-Rand des Live-Streams. Eine weitere gepunktete Linie, die den auf eingestellten Wert für die Funktionsunterdrückung darstellt **00:45:00**, befindet sich 45 Minuten früher in der Timeline als die gepunktete Linie für den Live-Rand. Der Zeitraum von 45 Minuten zwischen den gepunkteten Linien steht für den Zeitraum der Verfügungsunterbindung. Zu Beginn des

Zeitraums zur Verfügungsunterbrechung ist eine Werbeunterbrechung im Gange. Zwei weitere Werbeunterbrechungen treten während des Zeitraums der Verfügbarkeitsunterbrechung auf. Wie in der Abbildung dargestellt, werden alle Werbeunterbrechungen, die innerhalb des Verfügbarkeitsunterdrückungszeitraums auftreten **BEHIND_LIVE_EDGE**, MediaTailor personalisiert, wenn der Verfügbarkeitsunterdrückungsmodus auf und der Wert für die Verfügbarkeitsunterbindung auf „00:45:00“ hinter dem Live-Edge“ eingestellt ist. MediaTailor personalisiert nicht die Werbeunterbrechung, die zu Beginn des Zeitraums zur Verfügungsunterbindung gerade läuft.

Avail suppression mode: **BEHIND_LIVE_EDGE**

Avail suppression value: **00:45:00**



Example 4: **AFTER_LIVE_EDGE** Werbeunterbrechung ohne Werbeunterbrechungen während der Dauer der Verfügbarkeitsunterbrechung

Wenn der Verfügbarkeitsunterdrückungsmodus auf eingestellt ist **AFTER_LIVE_EDGE** und der Wert für die Verfügbarkeitsunterbindung größer als Null ist, werden Werbeunterbrechungen erst personalisiert, wenn die verstrichene Zeit der Sitzung diesen Wert erreicht MediaTailor hat.

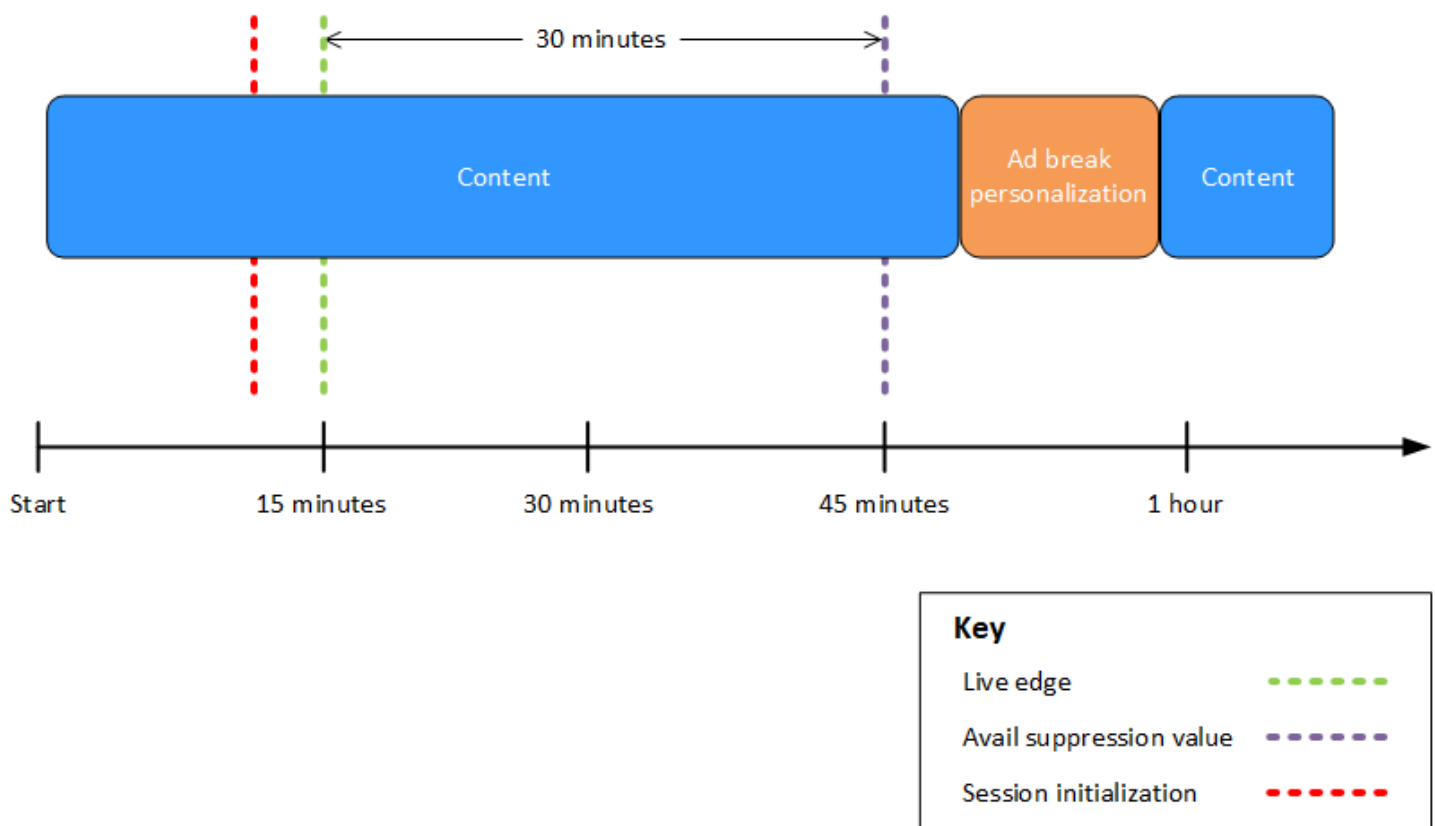
In der folgenden Abbildung sind verschiedene Blöcke horizontal entlang einer Zeitleiste angeordnet, die von links nach rechts verläuft. Jeder Block steht für einen Zeitraum, in dem der Inhalt des

Livestreams oder einer personalisierten Werbepause wiedergegeben wird. Eine gepunktete Linie steht für den aktuellen Live-Rand des Live-Streams. Eine weitere gepunktete Linie, die den eingestellten Wert für die Funktionsunterdrückung darstellt 00:30:00, erscheint 30 Minuten später in der Timeline in Bezug auf die gepunktete Linie für den Live-Rand. Eine dritte gepunktete Linie, die für die Sitzungsinitialisierung steht, befindet sich in der Timeline an einem anderen Punkt als die gepunktete Linie für die Live-Kante. Der Zeitraum von 30 Minuten zwischen der Live-Edge-Zeit und der Uhrzeit stellt den Zeitraum für die Unterdrückung der avail-suppression-value Nutzung dar. Nach Ablauf der Verfügbarkeitsunterbrechung erfolgt eine Werbeunterbrechung.

Wie in der Abbildung dargestellt, wird, wenn der Avail-Unterdrückungsmodus auf eingestellt ist AFTER_LIVE_EDGE, der Avail-Unterdrückungswert auf „00:30:00 nach dem Live Edge“ gesetzt, und die Sitzungsinitialisierung erfolgt vor dem Live-Edge, wodurch alle Werbeunterbrechungen MediaTailor personalisiert werden, die nach dem Verfügbarkeitsunterdrückungszeitraum auftreten.

Avail suppression mode: AFTER_LIVE_EDGE

Avail suppression value: 00:30:00



Example 5: **AFTER_LIVE_EDGE** Werbeunterbrechung mit der Option „**PARTIAL_AVAIL**Ausfüllen“ und einer aktiven Werbeunterbrechung am Ende des Zeitraums

Wenn der Modus zur Verfügungsunterbindung auf eingestellt ist **AFTER_LIVE_EDGE** und der Wert für die Verfügbarkeitsunterbindung größer als Null ist, werden Werbeunterbrechungen erst personalisiert, wenn die verstrichene Zeit der Sitzung diesen Wert erreicht MediaTailor hat.

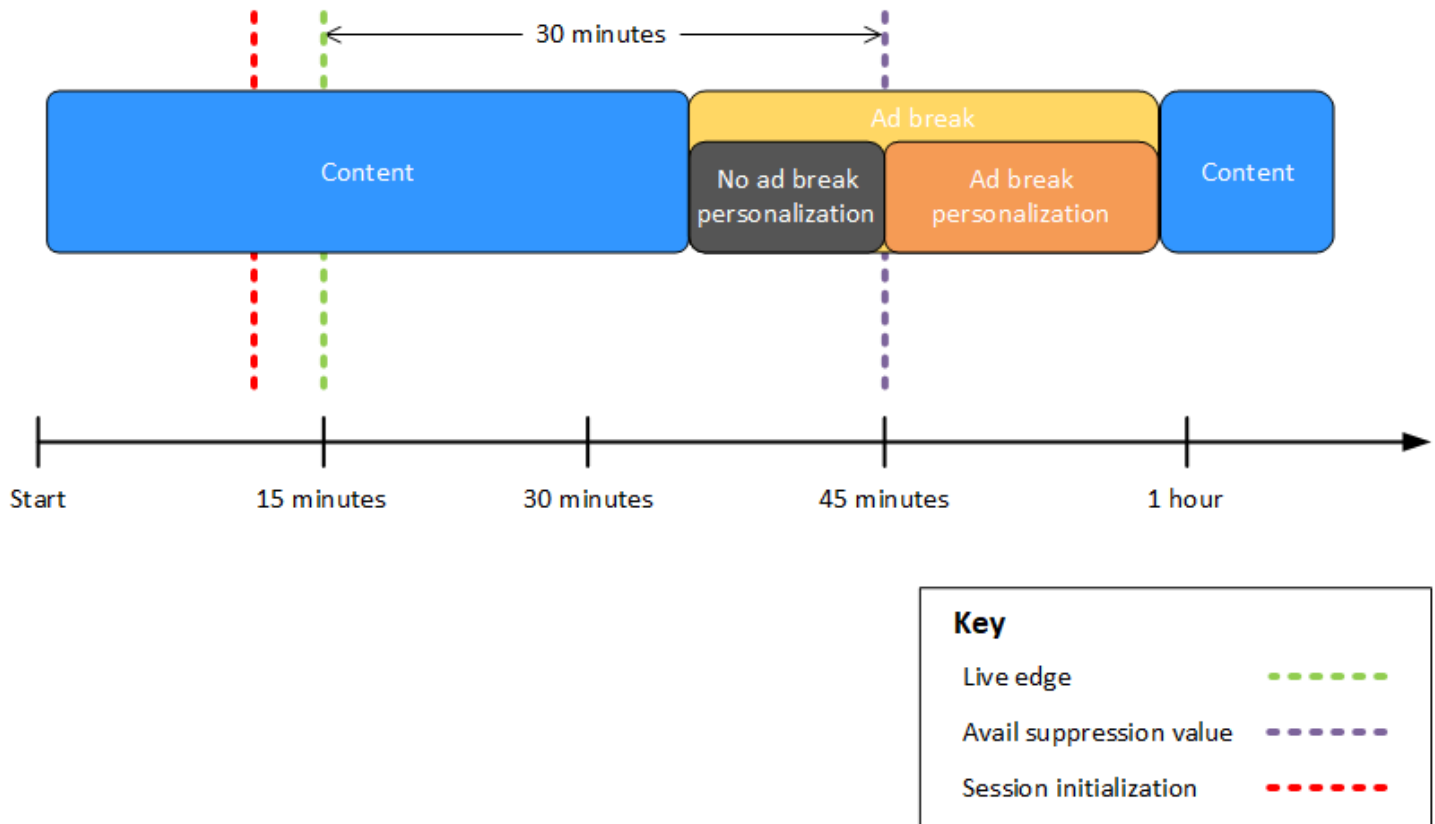
In der folgenden Abbildung sind verschiedene Blöcke horizontal entlang einer Zeitleiste angeordnet, die von links nach rechts verläuft. Jeder Block steht für einen Zeitraum, in dem der Inhalt eines Live-Streams, einer personalisierten Werbepause oder einer nicht personalisierten Werbepause abgespielt wird. Eine gepunktete Linie steht für den aktuellen Live-Rand des Live-Streams.

Eine weitere gepunktete Linie, die den eingestellten Wert für die Funktionsunterdrückung darstellt `00:30:00`, erscheint 30 Minuten später in der Timeline in Bezug auf die gepunktete Linie für den Live-Rand. Eine dritte gepunktete Linie, die für die Sitzungsinitialisierung steht, befindet sich in der Timeline an einem anderen Punkt als die gepunktete Linie für die Live-Kante. Der Zeitraum von 30 Minuten zwischen der Live-Edge-Zeit und der Uhrzeit stellt den Zeitraum für die Unterdrückung der `avail-suppression-value` Nutzung dar. Am Ende der Verfügbarkeitsunterbrechung wird eine Werbeunterbrechung ausgeführt. Wie in der Abbildung dargestellt, wird bei aktiviertem Verfügbarkeitsunterdrückungsmodus der Wert für die Verfügbarkeitsunterdrückung auf die Zeit `00:30:00` nach dem Live-Edge gesetzt, die Richtlinie zum Füllen der Verfügbarkeitsunterbrechung ist auf eingestellt und die Sitzungsinitialisierung erfolgt vor dem Live-Edge **PARTIAL_AVAIL**, wodurch alle Werbeunterbrechungen, die nach dem Verfügbarkeitsunterdrückungszeitraum auftreten, MediaTailor personalisiert werden. **AFTER_LIVE_EDGE** MediaTailor Personalisiert für die laufende Werbeunterbrechung am Ende des Verfügbarkeitsunterdrückungszeitraums den Teil dieser Werbeunterbrechung, der nach dem Verfügbarkeitsunterdrückungszeitraum erfolgt, personalisiert jedoch nicht den Teil dieser Werbeunterbrechung, der während des Verfügbarkeitsunterdrückungszeitraums stattfindet.

Avail suppression mode: **AFTER_LIVE_EDGE**

Avail suppression value: **00:30:00**

Avail suppression fill policy: **PARTIAL_AVAIL**



Example 6: **AFTER_LIVE_EDGE** Werbeunterbrechung mit der Richtlinie „**PARTIAL_AVAIL**Ausfüllen“ und einer aktiven Werbeunterbrechung von vor der Sitzungsinitialisierung bis nach dem Ende der Nutzungsunterbrechung

Wenn der Verfügbarkeitsunterdrückungsmodus auf eingestellt ist **AFTER_LIVE_EDGE** und der Wert für die Verfügbarkeitsunterbindung größer als Null ist, werden Werbeunterbrechungen erst personalisiert, wenn die Dauer der Sitzung diesen Wert erreicht MediaTailor hat.

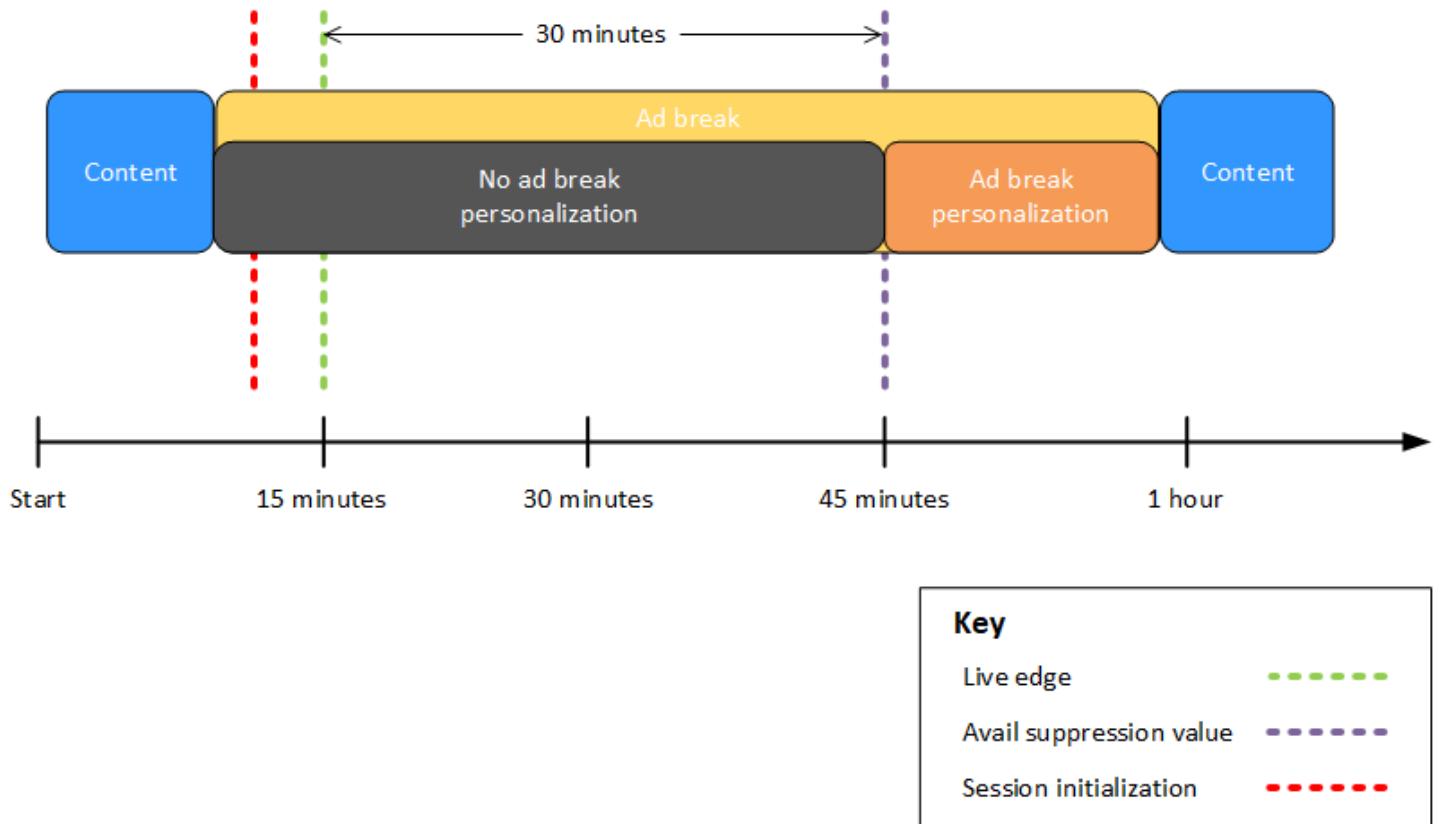
In der folgenden Abbildung sind verschiedene Blöcke horizontal entlang einer Zeitleiste angeordnet, die von links nach rechts verläuft. Jeder Block steht für einen Zeitraum, in dem der Inhalt eines Live-Streams, einer personalisierten Werbepause oder einer nicht personalisierten Werbepause abgespielt wird. Eine gepunktete Linie steht für den aktuellen Live-Rand des Live-Streams. Eine weitere gepunktete Linie, die den eingestellten Wert für die Funktionsunterdrückung darstellt **00:30:00**, erscheint 30 Minuten später in der Timeline

in Bezug auf die gepunktete Linie für den Live-Rand. Eine dritte gepunktete Linie, die für die Sitzungsinitialisierung steht, befindet sich in der Timeline an einem anderen Punkt als die gepunktete Linie für die Live-Kante. Der Zeitraum von 30 Minuten zwischen der Live-Edge-Zeit und der Uhrzeit stellt den Zeitraum für die Unterdrückung der avail-suppression-value Nutzung dar. Eine Werbeunterbrechung wird von einem Zeitpunkt vor der Sitzungsinitialisierung bis zu einem Zeitpunkt nach dem Zeitraum nach dem Verfügbarkeitsunterdrückungszeitraum ausgeführt. Wie in der Abbildung dargestellt, wird, wenn der Avail-Unterdrückungsmodus auf eingestellt ist `AFTER_LIVE_EDGE`, der Wert für die Avail-Unterdrückung auf die Zeit `00:30:00` nach dem Live-Edge gesetzt, die Richtlinie zum Füllen der Avail-Unterdrückung auf eingestellt ist und die Sitzungsinitialisierung vor dem Live-Edge erfolgt `PARTIAL_AVAIL`, MediaTailor personalisiert alle Werbeunterbrechungen, die nach dem Zeitraum der Verfügbarkeitsunterdrückung auftreten. MediaTailor Personalisiert für die Werbeunterbrechung, die vor, während und nach der Verfügbarkeitsunterdrückungszeit ausgeführt wird, den Teil dieser Werbeunterbrechung, der nach dem Verfügbarkeitsunterdrückungszeitraum erfolgt, personalisiert jedoch nicht den Teil dieser Werbeunterbrechung, der vor oder während des Verfügbarkeitsunterdrückungszeitraums stattfindet.

Avail suppression mode: **AFTER_LIVE_EDGE**

Avail suppression value: **00:30:00**

Avail suppression fill policy: **PARTIAL_AVAIL**



Example 7: **AFTER_LIVE_EDGE** Werbeunterbrechung, wenn zu Beginn des Zeitraums zur Unterdrückung der Verfügbarkeit eine Werbeunterbrechung im Gange ist

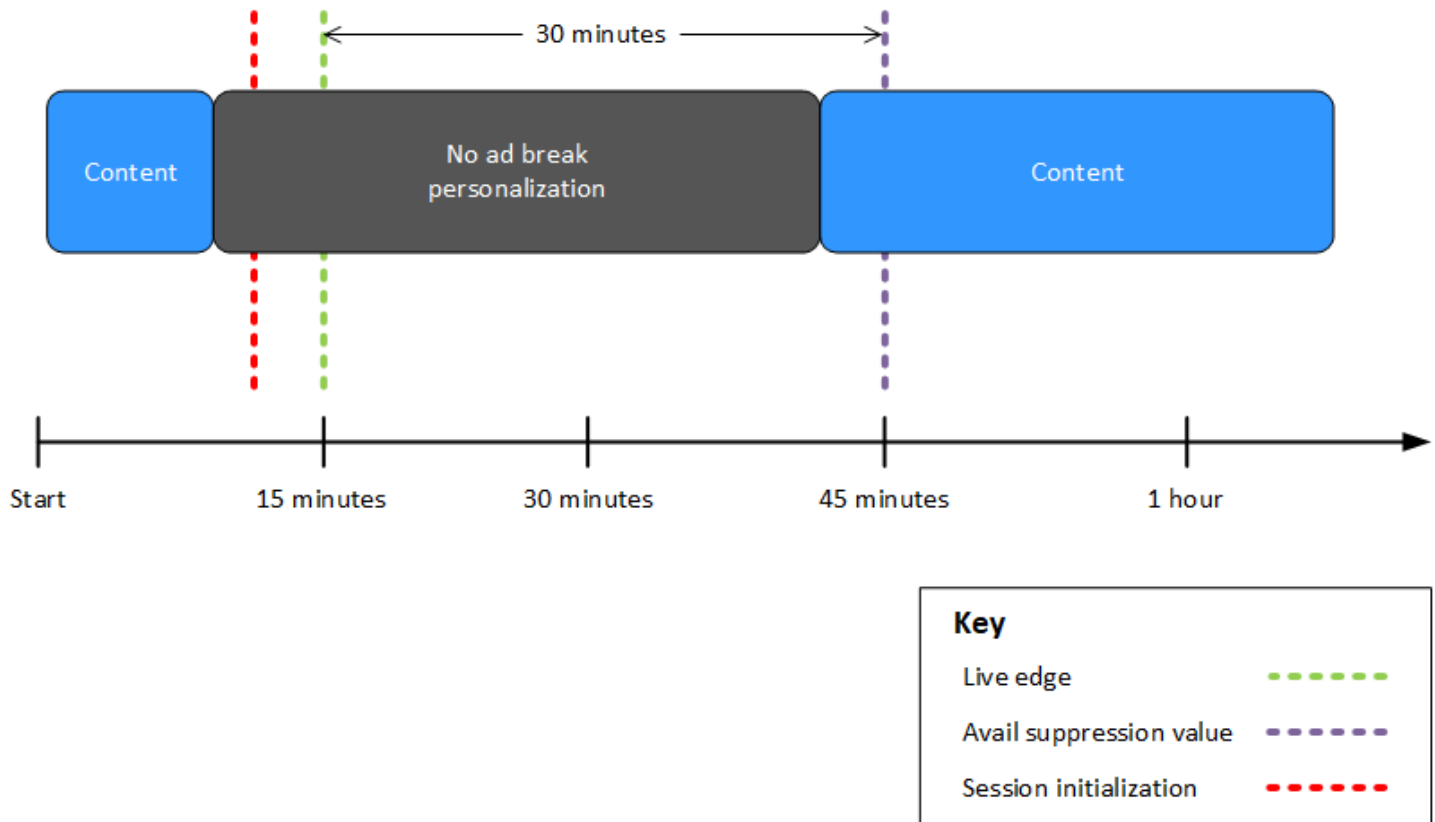
Wenn der Verfügbarkeitsunterdrückungsmodus auf eingestellt ist **AFTER_LIVE_EDGE** und der Wert für die Verfügbarkeitsunterbindung größer als Null ist, werden Werbeunterbrechungen erst personalisiert, wenn die verstrichene Zeit der Sitzung diesen Wert erreicht MediaTailor hat.

In der folgenden Abbildung sind verschiedene Blöcke horizontal entlang einer Zeitleiste angeordnet, die von links nach rechts verläuft. Jeder Block steht für einen Zeitraum, in dem der Inhalt des Live-Streams oder einer nicht personalisierten Werbepause wiedergegeben wird. Eine gepunktete Linie steht für den aktuellen Live-Rand des Live-Streams. Eine weitere gepunktete Linie, die den eingestellten Wert für die Funktionsunterdrückung darstellt **00:30:00**, erscheint 30 Minuten später in der Timeline in Bezug auf die gepunktete Linie für den Live-Rand. Eine dritte gepunktete Linie, die für die Sitzungsinitialisierung steht, befindet sich in der Timeline an einem anderen Punkt

als die gepunktete Linie für die Live-Kante. Der Zeitraum von 30 Minuten zwischen der Live-Edge-Zeit und der Uhrzeit stellt den Zeitraum für die Unterdrückung der avail-suppression-value Nutzung dar. Eine Werbeunterbrechung läuft von einem Zeitpunkt vor der Sitzungsinitialisierung bis zu einem Zeitpunkt innerhalb des Zeitraums der Verfügbarkeitsunterbrechung. Wie in der Abbildung dargestellt, wird, wenn der Avail-Unterdrückungsmodus auf eingestellt ist `AFTER_LIVE_EDGE`, der Wert für die Avail-Unterdrückung auf `00:30:00` nach dem Live-Edge gesetzt, und die Sitzungsinitialisierung erfolgt vor der Live-Edge-Zeit, aber nach dem Beginn der Werbeunterbrechung, MediaTailor wodurch diese Werbeunterbrechung nicht personalisiert wird.

Avail suppression mode: `AFTER_LIVE_EDGE`

Avail suppression value: `00:30:00`



Konfigurieren von Werbeunterdrückungsparametern – Wiedergabe-Sitzungsanforderung

Sie können die Einstellungen für die Anzeigenunterdrückung über Parameter in Ihrer ersten server- oder clientseitigen Wiedergabe-Sitzungsanforderung an konfigurieren. MediaTailor Wenn Sie die Einstellungen zur Werbeunterdrückung bereits über die MediaTailor Konsole oder AWS Elemental MediaTailor API konfiguriert haben, haben diese Parameter Vorrang vor diesen Einstellungen.

Sowohl der Werbeplatz-Unterdrückungsmodus als auch der Werbeplatz-Unterdrückungswert sind erforderlich, damit die Werbeunterdrückung funktioniert. Diese Parameter können nicht aus verschiedenen Quellen konfiguriert werden. Sie können beispielsweise nicht einen Parameter mit der MediaTailor Konsole und einen anderen mit einem Abfrageparameter konfigurieren.

MediaTailor unterstützt die folgenden Parameter zur Anzeigenunterdrückung.

Name	Beschreibung	Akzeptierte Werte
<code>availSuppressionMode</code>	Legt den Modus für die Werbeunterdrückung fest. Die Standardeinstellung für die Anzeigenunterdrückung ist <code>OFF</code> . Wenn diese Option auf <code>BEHIND_LIVE_EDGE</code> eingestellt ist, werden Werbeunterbrechungen nicht <code>aws.availSuppressionValue</code> pünktlich oder nachträglich aufgefüllt. Wenn diese Option auf <code>AFTER_LIVE_EDGE</code> eingestellt ist, werden Werbeunterbrechungen während oder nach Ablauf der Sperrzeit nicht gefüllt. Der Zeitraum für die Unterdrückung der Verfügbarkeit reicht von der Live-Edge-Zeit bis zu diesem Zeitpunkt, zuzüglich zusätzlicher <code>aws.availSuppressionValue</code> Pufferzeit.	<code>OFF</code> <code>BEHIND_LIVE_EDGE</code> <code>AFTER_LIVE_EDGE</code>
<code>availSuppressionValue</code>	Ein Zeitpunkt relativ zur Live-Grenze in einem Livestream.	Ein URL-codierter UTF-8-Zeitcode in. <code>HH:MM:SS</code> Zum Beispiel wären 1 Stunde und 30 Minuten <code>01%3A30%3A00</code> .
<code>availSuppressionFullPolicy</code>	Definiert die Richtlinie, die auf den Avail-Unterdrückungsmodus angewendet werden soll. <code>BEHIND_LIVE_EDGE</code> verwendet immer die Richtlinie zur Unterdrückung des vollen Nutzens. <code>AFTER_LIVE_EDGE</code> kann verwendet werden, um das teilweise Auffüllen von Werbeunterbrechungen	<code>PARTIAL_Avail</code> - nicht verfügbar für den <code>BEFORE_LIVE_EDGE</code>

Name	Beschreibung	Akzeptierte Werte
	aufzurufen, wenn eine Sitzung mitten in der Pause beginnt.	Unterdrückungsmodus • FULL_AVAI L_ONLY - der Standardwert für den AFTER_LIVE_EDGE Unterdrückungsmodus

Serverseitige Konfiguration

Der Basis-Abfrageparameter ist `aws.availSuppression`, gefolgt von optionalen Parameternamen/Werte-Paaren. Um die Abfrage zu erstellen, fügen Sie am Ende `aws.availSuppression=` der Wiedergabesitzung eine Anfrage an MediaTailor, gefolgt von Parameternamen und -werten. Weitere Hinweise zum Erstellen einer serverseitigen Wiedergabesitzungsanforderung finden Sie unter [Serverseitiges Ad-Tracking](#).

Beispiel: HLS

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/index.m3u8?
aws.availSuppressionMode=BEHIND_LIVE_EDGE&aws.availSuppressionValue=00%3A00%3A21
```

Die serverseitige Abfragesyntax ist in der folgenden Tabelle aufgeführt.

Abfragezeichenfolgekomponente	Beschreibung
?	Ein eingeschränktes Zeichen, das den Anfang einer Abfrage markiert.
aws.	Die Basisabfrage, gefolgt von Parametern aus Name- und Wert-Paaren. Eine Liste aller verfügbaren Parameter finden Sie unter Konfigurieren von Werbeunterdrückungsparametern – Wiedergabe-Sitzungsanforderung .

Abfragezeichenfolge enkomponente	Beschreibung
=	Ordnet den Parameternamen einem Wert zu. Beispiel, <code>aws.availSuppressionMode= BEHIND_LIVE_EDGE</code> .
&	Verkettet Abfrageparameter. Beispiel: <code>aws.availSuppressionMode= BEHIND_LIVE_EDGE</code> und <code>aws.availSuppressionValue= 00:30:00&aws.availSuppressionFillPolicy= FULL_AVAIL_ONLY</code> >.

Clientseitige Konfiguration

Nehmen Sie `availSuppression` Parameter in die POST-Anfrage Ihres Kunden auf MediaTailor. Weitere Hinweise zum Erstellen einer clientseitigen Wiedergabebesitzungsanforderung finden Sie unter [Kundenseitiges Ad-Tracking](#).

Beispiel: HLS

```
POST parent.m3u8
{
  "availSuppression": {
    "mode": "BEHIND_LIVE_EDGE",
    "value": "00:00:21",
    "fillPolicy": "FULL_AVAIL_ONLY"
  }
}
```

Stoßstangen einsetzen

Bumper sind kurze Video- oder Audioclips, die nicht übersprungen werden können und zu Beginn oder vor dem Ende einer Werbepause abgespielt werden.

Für Bumper gelten die folgenden Bedingungen:

- Die Länge der Stoßstangen darf 10 Sekunden oder weniger betragen.
- Bumper können zu Beginn einer Werbeunterbrechung, direkt vor dem Ende einer Werbeunterbrechung oder an beiden Stellen eingefügt werden.

- Bumper werden bei jeder Werbeunterbrechung in einer Playback-Sitzung abgespielt, sofern Pre-Roll nicht konfiguriert ist. Wenn Pre-Roll konfiguriert ist, werden Bumper während der Pre-Roll-Pause nicht abgespielt. Stattdessen werden sie in jeder nachfolgenden Pause nach dem Pre-Roll abgespielt.
- Für HLS müssen Sie das `duration` Attribut jedem EXT-X-CUE-OUT SCTE-35-Tag beifügen.
- Bumper werden so transcodiert, dass sie dem Quellinhalt entsprechen.
- Für Bumper werden Ihnen keine Gebühren berechnet.

Stoßstangen konfigurieren

Um Bumper zu verwenden, konfigurieren Sie den Bumper URLs mit der MediaTailor Konsole, der MediaTailor API oder der AWS Command Line Interface ().AWS CLI Sie können einen Start-Bumper, einen End-Bumper oder beides konfigurieren. Bumper werden auf einem Server wie Amazon Simple Storage Service (Amazon S3) gespeichert. Die Bumper URLs geben den Standort der gespeicherten Bumper-Asset (s) an.

Beispiel für Start- und Endstoßstange: URLs

URL für den Start-Bumper: `https://s3.amazonaws.com/startbumperad`

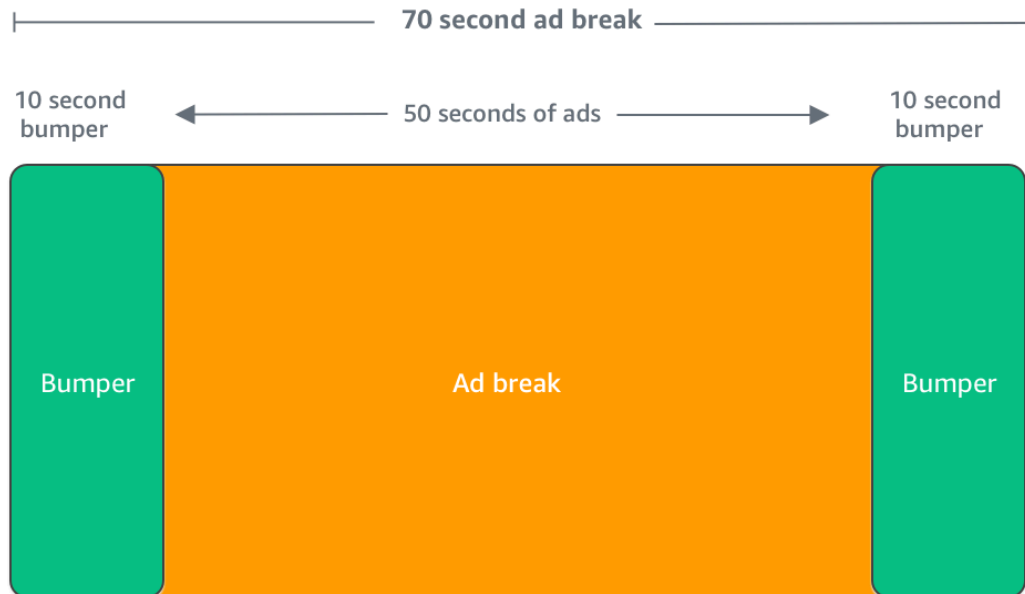
URL für das Ende des Bumpers: `https://s3.amazonaws.com/endbumperad`

Beispiel

Im Folgenden finden Sie ein Beispiel für das Verhalten von Bumper-Anzeigen.

Example 1: Bumper am Anfang und Ende

In diesem Beispiel sind die Start- und Endstoßstangen aktiviert. Der Ad Decision Server verfügt über 50 Sekunden mit personalisierten Anzeigen, um eine 70-sekündige Werbeunterbrechung zu füllen. Der 10-Sekunden-Start-Bumper wird zu Beginn der Werbepause abgespielt, 50 Sekunden lang werden Anzeigen abgespielt, danach der 10-Sekunden-End-Bumper.



Pre-Roll-Anzeigen einfügen

Note

Konfigurierbare Pre-Roll-Anzeigen sind nur für Live-Workflows verfügbar. Einzelheiten dazu, wie das Einfügen von Anzeigen (einschließlich Pre-Roll) für VOD funktioniert, finden Sie unter [Verhalten beim Zusammenfügen von Anzeigen bei VOD](#)

MediaTailor kann Anzeigen zu Beginn einer Playback-Sitzung einfügen, bevor der Hauptinhalt beginnt. Dies sind Pre-Roll-Werbespots.

Um Pre-Roll-Werbung einzufügen, füllen Sie die Felder Live pre-roll ad decision server (Live Pre-Roll-Ad Decision Server) und Live pre-roll maximum allowed duration (Maximal zulässige Live-Pre-Roll-Dauer) in den Einstellungen unter Additional (Zusätzlich) für Ihre Konfiguration aus, wie unter [Optionale Konfigurationseinstellungen](#) beschrieben.

1. Wenn es eine Wiedergabeanforderung MediaTailor erhält, sendet es anhand der folgenden Felder in der MediaTailor Wiedergabekonfiguration eine Anfrage für Pre-Roll-Anzeigen an:
 - Der Live-Pre-Roll-Anzeigenentscheidungsserver ist die URL des Ad Decision Servers (ADS), MediaTailor an den die Anfrage für Pre-Roll-Anzeigen gesendet wird.
 - Live pre-roll maximum allowed duration (Maximal zulässige Live-Pre-Roll-Dauer) ist die maximale Dauer für die Pre-Roll-Werbung. MediaTailor führt die folgende Aktion basierend auf der maximal zulässigen Dauer aus:
 - Wenn die Gesamtdauer der Anzeigen in der ADS-Antwort unter dem Wert liegt, den Sie im Feld Höchstdauer für Live-Pre-Roll angegeben haben, werden alle Anzeigen MediaTailor eingefügt. Wenn die letzte Anzeige fertig ist, wird MediaTailor sofort zum zugrunde liegenden Inhalt zurückgekehrt.
 - Wenn die Gesamtdauer der Anzeigen in der ADS-Antwort den Wert übersteigt, den Sie im Feld Höchstdauer des Live-Pre-Rolls angegeben haben, MediaTailor wählt eine Gruppe von Anzeigen aus, die in die Dauer passen, ohne sie zu überschreiten. MediaTailor fügt diese Anzeigen ohne Kürzung oder Kürzung ein. MediaTailor kehrt zum zugrunde liegenden Inhalt zurück, wenn die zuletzt ausgewählte Anzeige abgeschlossen ist.
2. Wenn es die Pre-Roll-Antwort von ADS MediaTailor erhält, manipuliert es das Manifest, um Links zu den Pre-Roll-Anzeigen hinzuzufügen. MediaTailor berechnet die Startzeit der Pre-Roll-Werbepause wie folgt:
 - Für DASH lautet die Formel $(publishTime - availabilityStartTime) - \max(suggestedPresentationDelay, minBufferTime)$.
 - Für HLS lautet die Formel $\max(2 * EXT-X-TARGETDURATION, EXT-X-START:TIMEOFFSET)$.
3. MediaTailor bestimmt, welche Maßnahmen bei Werbeunterbrechungen ergriffen werden sollen, bei denen es sich nicht um Pre-Rolls handelt. Wenn sich die Pre-Roll mit einer anderen Werbepause überschneidet, MediaTailor wird der überlappende Teil der Werbepause nicht personalisiert.

Schiefer wird eingefügt

Note

Slate ist nur für Live-Workflows verfügbar.

Mit AWS Elemental MediaTailor können Sie eine Slate-Anzeige für Werbeunterbrechungen festlegen. Ein Slate ist ein MP4 Standard-Asset, das in einen Stream eingefügt wird, z. B. ein Standbild oder ein Video in einer Endlosschleife, der anstelle des Live-Inhalts abgespielt wird.

AWS Elemental MediaTailor zeigt in den folgenden Situationen ein Slate an:

- Um einen Zeitraum auszufüllen, die durch eine Werbeersetzung nicht vollständig gefüllt wird
- Wenn der ADS (Ad Decision Server) mit einer leeren VAST- oder VMAP-Antwort antwortet
- Bei Fehlerbedingungen, z B. ADS-Timeout
- Wenn die Dauer der Werbung länger als die Werbepause ist
- Wenn keine Werbung verfügbar ist

Wenn Sie kein Slate konfigurieren, MediaTailor wird standardmäßig der zugrunde liegende Inhaltsstream verwendet, wenn eine der oben genannten Bedingungen erfüllt ist.

Das Slate konfigurieren

Sie bestimmen das Slate im Bereich für zusätzliche Konfigurationen in der [MediaTailorKonsole](#). MediaTailor lädt das Slate von der von Ihnen angegebenen URL herunter und transkodiert es in dieselben Wiedergabeversionen wie Ihr Inhalt. Über die optionale Konfiguration des Schwellenwerts für Personalisierung in der Konsole können Sie festlegen, wie lange ein Slate maximal angezeigt wird. MediaTailor Weitere Informationen finden Sie unter [the section called “Optionale Konfigurationseinstellungen”](#).

Slate-Konfiguration und VPAID

Sie müssen ein Slate konfigurieren, wenn Sie VPAID-Anzeigen verwenden. Um Platz für VPAID-Anzeigen zu haben, die der Videoplayer einfügt, MediaTailor fügt das Slate für die Dauer der VPAID-Anzeige ein. Diese Dauer kann etwas länger sein als die Dauer der VPAID-Anzeige, wie von VAST gemeldet, um der Nutzerinteraktivität Rechnung zu tragen, wie unter beschrieben. [VPAID-Anforderungen](#) Der Videoplayer verarbeitet die VPAID-Anzeige dann auf der Grundlage der zurückgegebenen clientseitigen Berichtsmetadaten. MediaTailor Weitere Informationen zur clientseitigen Berichterstellung finden Sie unter [the section called “Kundenseitiges Tracking”](#).

Anzeigen vorab abrufen

Verwenden Sie den AWS Elemental MediaTailor Anzeigenvorabruf für Livestreams, um die Spitzenlast auf Ad Decision Servern (ADS) zu reduzieren und die Latenz bei der Manifestzustellung

zu Beginn jeder Werbeunterbrechung zu verringern. Wenn Sie einen Zeitplan für den Vorabruf definieren, MediaTailor folgt dieser Zeitplan, um Anzeigen aus dem ADS abzurufen und sie für die Anzeigenschaltung vorzubereiten, bevor sie für eine Werbeunterbrechung benötigt werden. Bei Livestreams kann das Prefetching dazu beitragen, verringerte Anzeigenfüllraten und verpasste Monetarisierungsmöglichkeiten aufgrund von Timeouts bei Anzeigenanfragen und Transcodierung oder anderen Netzwerkverzögerungen zu vermeiden.

Um den Vorabruf von Anzeigen einzurichten, erstellen Sie in Ihrer Wiedergabekonfiguration einen oder mehrere Prefetch-Zeitpläne. Ein Prefetch-Zeitplan gibt an, MediaTailor wie und wann Werbeanzeigen abgerufen und für eine bevorstehende Werbepause vorbereitet werden müssen.

- Wenn für eine Veranstaltung Werbeanzeigen verfügbar sind, die nach einem vorhersehbaren Zeitplan verfügbar sind, verwenden Sie einen einzigen Prefetch-Zeitplan. Jeder einzelne Prefetch-Zeitplan definiert eine einzelne Gruppe von Anzeigen, die in einer einzigen Anzeigenauswahl platziert MediaTailor werden sollen. Um Anzeigen für mehrere verfügbare Anzeigen vorab abzurufen, wenn Sie einzelne Prefetch-Zeitpläne verwenden, müssen Sie mehrere Prefetch-Zeitpläne (bis zu 24 Stunden vor der Anzeigenbereitstellung) erstellen, die für jede verfügbare Anzeige relevant sind.
- Wenn für eine Veranstaltung Werbeanzeigen verfügbar sind, die nicht in einem vorhersehbaren Zeitplan verfügbar sind, verwenden Sie einen wiederkehrenden Prefetch-Zeitplan. Ein wiederkehrender Prefetch-Zeitplan erstellt automatisch einen Zeitplan und ruft Anzeigen vor jeder Werbeunterbrechung in einem Event vorab ab. Im Zeitplan für wiederkehrende Prefetches werden Anzeigen für jede verfügbare Anzeige innerhalb eines bestimmten Zeitraums abgerufen (bis zu 24 Stunden vor Ende der Veranstaltung). Sie müssen nicht für jede verfügbare Anzeige einen Zeitplan erstellen, verlieren jedoch einen Teil der zeitlichen Kontrolle, die ein einzelner Prefetch-Vorgang bietet.

In den folgenden Themen wird der Vorabruf von Anzeigen näher beschrieben.

Themen

- [So funktioniert der Vorabruf](#)
- [Prefetch-Zeitpläne erstellen](#)
- [Löschen von Prefetch-Zeitplänen](#)

So funktioniert der Vorabruf

Wenn Ihr Client eine Manifestanforderung an stellt MediaTailor, wertet der Dienst alle Prefetch-Zeitpläne aus, die mit der Wiedergabekonfiguration verknüpft sind. Wenn MediaTailor kein passender Prefetch-Zeitplan gefunden wird, kehrt der Dienst zur normalen Anzeigeneinfügung zurück und ruft keine Anzeigen vorab ab.

Wenn ein MediaTailor passender Prefetch-Zeitplan gefunden wird, bewertet der Dienst den Zeitplan anhand von zwei Komponenten: Abruf und Nutzung. Die Konfiguration der einzelnen Komponenten variiert zwischen einzelnen Prefetch-Zeitplänen und wiederkehrenden Prefetch-Zeitplänen, wie in den folgenden Abschnitten beschrieben.

Ablauf eines einzelnen Prefetch-Zeitplans

Abruf

Dadurch wird das Abrufenfenster definiert. Dabei handelt es sich um den Zeitraum, in dem Anzeigen MediaTailor vorab aus dem ADS abgerufen werden. Achten Sie darauf, dieses Fenster für einen Zeitraum vor der Werbeunterbrechung einzuplanen. Im Folgenden finden Sie einen Überblick darüber, wie ein einzelner Prefetch-Zeitplan MediaTailor verarbeitet wird.

Die Schritte zum Erstellen eines einzelnen Prefetch-Zeitplans in der Konsole finden Sie unter [Prefetch-Zeitpläne erstellen](#). Eine API-Anleitung finden Sie [PrefetchSchedules](#) in der AWS Elemental MediaTailor API-Referenz.

MediaTailor Sendet während des angegebenen Abrufenfensters Anfragen an das ADS, um Anzeigen abzurufen und für die spätere Einfügung in Wiedergabe-Sitzungen vorzubereiten.

- Wenn Sie das Traffic-Shaping-Fenster festlegen, werden Anfragen über die angegebene Anzahl von Sekunden MediaTailor verteilt, anstatt Anfragen für alle Sitzungen gleichzeitig zu senden. Diese verteilte Verteilung des Traffics trägt dazu bei, dass die ADS nicht überlastet werden, was zu Timeouts und niedrigen Anzeigenfüllraten führt.
- Wenn Sie dynamische Variablen einrichten, werden diese MediaTailor Variablen in die Anfragen an das ADS aufgenommen. MediaTailor verwendet diese Variablen, um Zeitpläne für den Vorabruf während des Nutzungsfensters abzugleichen und verfügbar zu machen. Weitere Informationen finden Sie im folgenden Abschnitt zum Verbrauch.

Example

Eine Live-Veranstaltung dauert von 7:45 Uhr bis 10 Uhr, mit einer Werbepause um 8:15 Uhr. Sie konfigurieren MediaTailor den Abruf von Werbeanzeigen von 7:45 Uhr bis 8 Uhr mit einem Traffic-

Shaping-Zeitfenster von 60 Sekunden. Bei 500.000 gleichzeitigen Benutzern werden die ADS-Anfragen so MediaTailor verteilt, dass eine durchschnittliche Rate von etwa 8.333 Transaktionen pro Sekunde für 60 Sekunden (500.000 Benutzer/60 Sekunden = 8.333 Anfragen pro Sekunde) erreicht wird, anstatt alle Anfragen gleichzeitig zu senden.

Die Abrufkonfiguration umfasst die dynamische Variable Schlüssel und Wert. `scte.event 1234` MediaTailor nimmt diese Variable in die Anfragen an die ADS auf, die dann verwendet werden können, um bestimmte Werbetreibende auf die Event-ID 1234 auszurichten.

Verbrauch

Wenn Sie während des Nutzungsfensters MediaTailor auf SCTE-35-Werbeunterbrechungsmarken stoßen, werden die vorab abgerufenen Anzeigen in einer Werbeunterbrechung platziert.

- Wenn Sie keine Auswahlkriterien festgelegt haben, werden Werbeanzeigen in der ersten Pause des Werbefensters MediaTailor eingefügt.
- Wenn Sie einen dynamischen Variablenschlüssel für verfügbare Übereinstimmungskriterien festgelegt haben, werden diese Kriterien anhand der dynamischen Variablen MediaTailor bewertet, die Sie im Abruffenster festgelegt haben. Eine Werbeunterbrechung kann nur dann vorab abgerufen werden, wenn die Auswahlkriterien erfüllt sind. MediaTailor fügt Anzeigen in die erste Pause ein, die die Kriterien erfüllt.

Eine Liste der unterstützten Kriterien für den Abgleich von Verfügbarkeiten finden Sie in der Tabelle unter In der Spalte Verfügbar für Anzeigenvorabruf. [Verwenden von Sitzungsvariablen](#)

Example fortgesetzt

Sie haben die Startzeit für den Konsum auf 8:15 Uhr und die Endzeit auf 8:17 Uhr festgelegt. Sie geben `scte.event_id` für den Schlüssel in die Verfügbarkeit die passenden Kriterien an.

Für jede Werbepause, die zwischen MediaTailor 8:15 Uhr und 8:17 Uhr angezeigt wird, wird Folgendes ausgewertet SCTE Event-ID für jede Werbeunterbrechung. MediaTailor Fügt in jeder Wiedergabe-Sitzung die vorab abgerufenen Anzeigen in die erste Werbeunterbrechung ein, die die Event-ID 1234 hat (wie in den dynamischen Variablen für den Abruf definiert). MediaTailorFührt bei Werbeunterbrechungen, die nicht die richtige Event-ID enthalten, die Standardanzeigeneinfügung durch.

Ablauf eines wiederkehrenden Prefetch-Vorabrufs

Abruf

Dadurch wird das Zeitfenster für wiederkehrende Abrufe definiert. Dabei handelt es sich um den Zeitraum, in dem Anzeigen für ein Live-Event MediaTailor vorab abgerufen und eingefügt werden (bis zu 24 Stunden). Im Folgenden finden Sie einen Überblick darüber, wie wiederkehrende MediaTailor Prefetch-Zeitpläne verarbeitet werden.

Schritte zum Erstellen eines wiederkehrenden Prefetch-Zeitplans in der Konsole finden Sie unter [Prefetch-Zeitpläne erstellen](#). Eine API-Anleitung finden Sie [PrefetchSchedules](#) in der AWS Elemental MediaTailor API-Referenz.

MediaTailor ruft während des angegebenen wiederkehrenden Prefetch-Fensters Anzeigen für ein Live-Event ab und fügt sie ein, das bis zu 24 Stunden dauert. Ruft nach jeder Werbeunterbrechung im Fenster MediaTailor automatisch Anzeigen für die nächste Werbeunterbrechung ab.

- Wenn Sie die Verzögerung nach Ende der Verfügbarkeit festlegen, wird die angegebene Zeit MediaTailor gewartet, bevor die nächste Anzeigengruppe für die nächste Werbepause abgerufen wird.
- Wenn Sie das Traffic-Shaping-Fenster festlegen, werden Anfragen über die angegebene Anzahl von Sekunden MediaTailor verteilt, anstatt Anfragen für alle Sitzungen gleichzeitig zu senden. Diese verteilte Verteilung des Traffics trägt dazu bei, dass die ADS nicht überlastet werden, was zu Timeouts und niedrigen Anzeigenfüllraten führt.
- Wenn Sie dynamische Variablen einrichten, werden diese MediaTailor Variablen in die Anfragen an das ADS aufgenommen. MediaTailor verwendet diese Variablen, um Zeitpläne für den Vorabruf während des Nutzungsfensters abzugleichen und verfügbar zu machen. Weitere Informationen finden Sie im folgenden Abschnitt zum Verbrauch.

Example

Eine Live-Veranstaltung dauert von 19 Uhr bis 20:45 Uhr, mit vier Werbeunterbrechungen in dieser Zeit. Die Werbeunterbrechungen haben keinen vorhersehbaren Zeitplan. Sie konfigurieren den wiederkehrenden Prefetch von 19:00 Uhr bis 20:45 Uhr mit einer Verzögerung von 10 Minuten und einem Traffic-Shaping-Fenster von 60 Sekunden. MediaTailor ruft nach jedem Zugriff Anzeigen für die nächste Werbepause ab. Zehn Minuten nach Ende der Verfügbarkeit MediaTailor beginnt das Senden von Abrufanfragen an das ADS. Bei einem Zeitfenster von 60 Sekunden und 500.000 gleichzeitigen Benutzern werden die ADS-Anfragen so MediaTailor

verteilt, dass eine durchschnittliche Rate von etwa 8.333 Transaktionen pro Sekunde für 60 Sekunden erreicht wird (500.000 Benutzer/60 Sekunden = 8.333 Anfragen pro Sekunde), anstatt alle Anfragen gleichzeitig zu senden.

Die `scte.event 1234` Abrufkonfiguration umfasst die dynamische Variable Schlüssel und Wert. MediaTailor nimmt diese Variable in die Anfragen an die ADS auf, die dann verwendet werden können, um bestimmte Werbetreibende auf die Event-ID 1234 auszurichten.

Verbrauch

Wenn MediaTailor auf SCTE-35-Markierungen für Werbeunterbrechungen gestoßen wird, werden die vorab abgerufenen Anzeigen in einem Werbeblock platziert.

- Wenn Sie das Ablaufdatum für abgerufene Anzeigen festlegen, können vorab abgerufene Anzeigen bis zum angegebenen Ablaufdatum eingefügt werden.
- Wenn Sie keine Auswahlkriterien für die Verfügbarkeit festgelegt haben, werden Anzeigen in der ersten Pause des Nutzungsfensters MediaTailor eingefügt.
- Wenn Sie einen dynamischen Variablenschlüssel für verfügbare Übereinstimmungskriterien festgelegt haben, werden diese Kriterien anhand der dynamischen Variablen MediaTailor bewertet, die Sie im Abruffenster festgelegt haben. Eine Werbeunterbrechung kann nur dann vorab abgerufen werden, wenn die Auswahlkriterien erfüllt sind. MediaTailor fügt Anzeigen in die erste Pause ein, die die Kriterien erfüllt.

Eine Liste der unterstützten Kriterien für den Abgleich von Verfügbarkeiten finden Sie in der Tabelle unter In der Spalte Verfügbar für Anzeigenvorabruf. [Verwenden von Sitzungsvariablen](#)

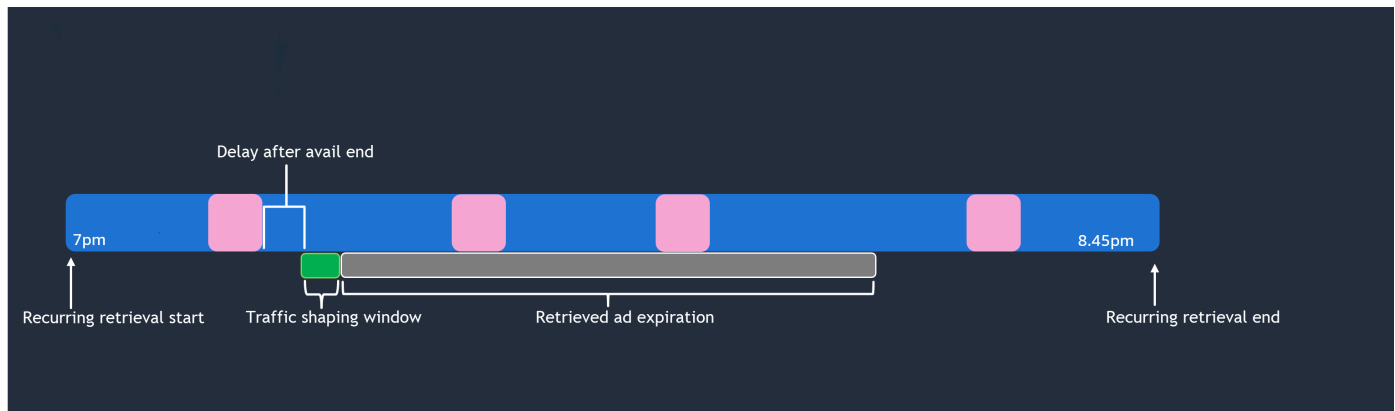
Example fortgesetzt

In den Verbrauch beziehen Sie `scte.event_id` für den Schlüssel in die Verfügbarkeit passende Kriterien mit ein.

Für jede Werbeunterbrechung, MediaTailor auf die Sie stoßen, wird Folgendes ausgewertet SCTE Event-ID für jede Werbeunterbrechung. MediaTailor Fügt in jeder Wiedergabe-Sitzung die vorab abgerufenen Anzeigen in jede Werbeunterbrechung ein, die die Event-ID 1234 hat (wie in den dynamischen Variablen für den Abruf definiert). MediaTailor Führt bei Werbeunterbrechungen, die nicht die richtige Event-ID enthalten, die Standardanzeigeneinfügung durch.

Sie haben den Ablauf der Anzeige auf 2700 Sekunden festgelegt, sodass die abgerufenen Anzeigen 45 Minuten lang eingeblendet werden können.

Die folgende Grafik veranschaulicht das Beispiel, wobei die kleinen Quadrate Werbeunterbrechungen darstellen. Die Einstellungen für den Zeitplan für wiederkehrende Prefetches werden entlang der Event-Zeitleiste dargestellt.



Grundlegendes zu den Kosten für das Prefetching

Für Anfragen zum Abrufen von Anzeigen fallen keine Kosten an. Für den Abruf von Prefetch-Anzeigen wird Ihnen jedoch der Standardtarif für die transkodierte Prefetch-Anzeigen berechnet. MediaTailor Für die Nutzung von Prefetch-Anzeigen, die in Werbeunterbrechungen platziert werden, wird Ihnen der Standardtarif für die Anzeigeneinblendung berechnet. MediaTailor [Informationen zu den Kosten für Transcodierung und Anzeigenschaltung finden Sie unter Preise.AWS Elemental MediaTailor](#)

Prefetch-Zeitpläne erstellen

Im folgenden Verfahren wird erklärt, wie Sie mithilfe der Konsole einen Prefetch-Zeitplan erstellen. MediaTailor Informationen zum programmgesteuerten Erstellen und Verwalten von Prefetch-Zeitplänen mithilfe der MediaTailor API finden Sie in der API-Referenz. [PrefetchSchedulesAWS Elemental MediaTailor](#)

Note

Bei der Konfiguration von Prefetch-Zeitplänen ist es wichtig zu verstehen MediaTailor, wie mit verschiedenen Variablentypen umgegangen wird.

Verwenden Sie passende Kriterien

Wenn Sie Avail-Abgleichskriterien in einem Zeitplan verwenden möchten, stellen Sie sicher, dass Sie zuerst die ADS-URL-Vorlage Ihrer Wiedergabekonfiguration mit

[dynamischen Sitzungsvariablen](#) konfigurieren, da die Auswahlkriterien sonst keine Wirkung haben. Informationen zum Arbeiten mit dynamischen Sitzungsvariablen finden Sie unter [Schritt 3: Konfigurieren Sie die ADS-Anforderungs-URL und die Abfrageparameter](#) dem Thema Erste Schritte beim Einfügen von MediaTailor Anzeigen.

Spielervariablen in Prefetch-Zeitplänen

Wenn Sie einen Prefetch-Zeitplan erstellen, definieren Sie in Ihrer Prefetch-Konfiguration keine Spielervariablen als dynamische Variablen. Übergeben Sie die Spielervariablen stattdessen so, wie Sie es normalerweise zu Beginn der Sitzung tun würden. MediaTailor schließt diese Variablen automatisch in Prefetch-Anzeigenanfragen ein, wenn die Variablen in der ADS-Vorlagen-URL zugeordnet sind.

Um einen neuen Prefetch-Zeitplan mit der Konsole zu erstellen

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>
2. Wählen Sie im Navigationsbereich Konfigurationen aus. Wählen Sie die Wiedergabekonfiguration aus, für die Sie einen Prefetch-Zeitplan erstellen möchten.
3. Wählen Sie auf der Registerkarte Prefetch-Zeitpläne die Option Prefetch-Zeitplan hinzufügen aus.
4. Gehen Sie im Bereich mit den Details zum Prefetch-Zeitplan wie folgt vor:
 - Geben Sie unter Name eine Kennung für Ihren Prefetch-Zeitplan ein, z. B. my-prefetch-schedule
 - Geben Sie als Stream-ID optional eine eindeutige ID ein. Wenn Ihr Origin mehrere Playback-Streams enthält, können Sie diese ID verwenden, um anzuweisen MediaTailor, Anzeigen in einem bestimmten Stream zu schalten. Wenn Ihre Wiedergabekonfiguration beispielsweise einen Sport-Stream und einen TV-Show-Stream umfasst, können Sie die Stream-ID verwenden, um Prefetch-Zeitpläne zu erstellen, um Werbeanzeigen einzufügen, die auf den Sport-Stream zugeschnitten sind. Sie übergeben den Stream-ID-Wert MediaTailor in der Sitzungsinitialisierungs- oder Manifestanforderung Ihres Kunden an. Weitere Informationen finden Sie im folgenden Beispiel.
 - Für serverseitiges Tracking fügen Sie den `?aws.streamId` Abfrageparameter und den Wert in die GET HTTP Anfrage Ihres Clients an Ihren MediaTailor Endpunkt ein. Allgemeine Informationen zur serverseitigen Nachverfolgung finden Sie unter [Serverseitiges Ad-](#)

[Tracking](#) Eine Manifestanforderung an einen HLS-Endpunkt, die eine Stream-ID enthält, sieht wie folgt aus, wobei der Name Ihrer Stream-ID *myStreamId* steht:

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/<asset-id>?
aws.streamId=myStreamId
```

- Für die clientseitige Nachverfolgung fügen Sie den `streamId` Schlüssel und den Wert in den Text der Anfrage zur POST HTTP Sitzungsinitialisierung Ihres Clients für den Endpunkt ein. MediaTailor/v1/session Allgemeine Informationen zur clientseitigen Nachverfolgung finden Sie unter. [Kundenseitiges Ad-Tracking](#) Eine Anfrage zur Sitzungsinitialisierung, die eine Stream-ID enthält, sieht wie folgt aus, wobei der Name Ihrer Stream-ID *myStreamId* steht:

```
POST <mediatailorURL>/v1/session/<hashed-account-id>/<origin-id>/<asset-id>
{
  'streamId': 'myStreamId'
}
```

5. Treffen Sie für den Typ Prefetch Ihre Auswahl und wählen Sie den entsprechenden Abschnitt aus, um Hilfe bei zusätzlichen Feldern zu erhalten:
 - Wählen Sie Single, wenn Sie einen Prefetch-Zeitplan für eine Werbepause in einer Veranstaltung erstellen.
 - Wählen Sie Wiederkehrend, wenn Sie einen Zeitplan erstellen, bei dem Anzeigen vor jeder Werbeunterbrechung einer Veranstaltung automatisch vorab abgerufen werden.

Einzelner Prefetch-Zeitplan

Um einen Zeitplan zu erstellen, bei dem Anzeigen vorab abgerufen werden, bevor eine Anzeige in einem Ereignis verfügbar ist.

1. Geben Sie im Bereich Abruf die Abruf-Einstellungen an, die Sie verwenden möchten. Diese Einstellungen bestimmen, wann Anzeigen aus dem ADS MediaTailor vorab abgerufen werden. Sie legen auch fest, welche dynamischen Sitzungsvariablen gegebenenfalls in die Anfrage an das ADS aufgenommen werden sollen.
 - Geben Sie unter Startzeit die Uhrzeit ein, zu der Prefetch-Abrufe für diese Werbeunterbrechung gestartet werden MediaTailor können. MediaTailor versucht, Anzeigen für Manifestanfragen, die Ihr Kunde an oder nach diesem Zeitpunkt gestellt hat, vorab

abzurufen. Der Standardwert ist die aktuelle Zeit. Wenn Sie keinen Wert angeben, beginnt der Dienst so bald wie möglich mit dem Prefetch-Abruf.

- Geben Sie im Feld Endzeit den Zeitpunkt ein, MediaTailor zu dem Sie das Vorabrufen von Anzeigen für diese Werbepause beenden möchten. MediaTailor versucht, Anzeigen für Manifestanfragen, die zu oder vor diesem Zeitpunkt erfolgen, vorab abzurufen. Das Abrufenfenster kann sich mit dem Nutzungsfenster überschneiden.
 - Geben Sie für die Dauer des Traffic Shaping-Fensters die Anzahl der Sekunden ein, über die Anfragen an das ADS verteilt werden MediaTailor sollen. Weitere Informationen finden Sie unter Erläuterung des [Zeitplans für einen einzelnen Prefetch-Abruf](#).
 - Geben Sie im Abschnitt [Dynamische Variablen](#) bis zu 100 dynamische Sitzungsvariablen ein. MediaTailor verwendet diese Variablen als Ersatz in Prefetch-Anfragen, die es an das ADS sendet. [Wenn Sie keine dynamischen Sitzungsvariablen eingeben, versucht es nach besten MediaTailor Kräften, die Werte für die in Ihrer ADS-URL enthaltenen dynamischen Variablen zu interpolieren.](#)
 - Wählen Sie Dynamische Variable hinzufügen aus.
 - Geben Sie als Schlüssel einen dynamischen Sitzungsvariablenschlüssel ein, z. `scte.event_id` B. Sie können jede dynamische Variable verwenden, die MediaTailor unterstützt. Hinweise zu dynamischen Sitzungsvariablen finden Sie unter [Verwenden von Sitzungsvariablen](#).
 - Geben Sie unter Wert einen dynamischen Variablenwert ein, z. *my-event* B.
 - Um eine weitere dynamische Variable hinzuzufügen, wählen Sie „Dynamische Variable hinzufügen“.
2. Geben Sie im Bereich Verbrauch die Einstellungen an, die Sie für das Verbrauchsfenster verwenden möchten. Diese Einstellungen bestimmen, wann MediaTailor die Anzeigen in den Werbeblock aufgenommen werden. Sie bestimmen auch alle verfügbaren Übereinstimmungskriterien, die Sie verwenden möchten.
- Geben Sie unter Startzeit die Uhrzeit ein, zu der Sie beginnen MediaTailor möchten, vorab abgerufene Anzeigen in der Werbepause zu platzieren. Der Standardwert ist die aktuelle Uhrzeit. Wenn Sie keine Uhrzeit angeben, beginnt der Dienst so bald wie möglich mit der Nutzung von Prefetches.
 - Geben Sie unter Endzeit einen Zeitpunkt ein, MediaTailor zu dem Sie die Platzierung der vorab abgerufenen Anzeigen in der Werbepause beenden möchten. MediaTailor versucht, Anzeigen für die Manifestanfragen Ihres Kunden, die zu oder vor diesem Zeitpunkt eingehen,

vorab abzurufen. Die Endzeit muss nach der Startzeit und in weniger als einem Tag liegen. Das Verbrauchsfenster kann sich mit dem Abruffenster überschneiden.

- Wählen [Sie im Abschnitt Verfügbare Kriterien](#) die Option Verfügbarkeitskriterien hinzufügen aus und fügen Sie Ihrem Zeitplan bis zu fünf passende Verfügbarkeitskriterien hinzu. Fügen Sie dann unter Dynamischer Variablenschlüssel einen dynamischen Variablenschlüssel hinzu, z. B. `scte.event_id`. MediaTailor platziert die vorab abgerufenen Anzeigen nur dann im Werbeblock, wenn sie die Kriterien erfüllen, die durch die dynamischen Variablenwerte definiert sind. MediaTailor, an die Ihr Kunde entweder weiterleitet oder die MediaTailor aus Informationen wie Sitzungsdaten abgeleitet werden. Wenn eine Werbeunterbrechung die angegebenen Übereinstimmungskriterien nicht erfüllt, wird der Vorabruf für diese Werbepause MediaTailor übersprungen. Weitere Informationen finden Sie in der Erläuterung zum Verbrauch [eines einzelnen Prefetch-Zeitplans](#).

3. Wählen Sie „Verfügbarkeitskriterien hinzufügen“ aus.

Prefetch-Zeitpläne laufen automatisch ab, wenn das Nutzungsfenster abgelaufen ist. Zu Diagnosezwecken bleiben sie mindestens 7 Tage lang sichtbar und MediaTailor werden danach automatisch gelöscht. Alternativ können Sie einen Prefetch-Zeitplan jederzeit manuell löschen. Informationen zum manuellen Löschen eines Prefetch-Zeitplans finden Sie im folgenden Abschnitt. [the section called “Löschen von Prefetch-Zeitplänen”](#)

Bestimmen Sie, wie oft Ihr Client die API aufrufen soll `CreatePrefetchSchedule`

Ihr Kunde kann die [CreatePrefetchSchedule](#) API einmal täglich programmgesteuert aufrufen, um den Abruf und die Nutzung einzurichten, sofern Sie genau wissen, wann Werbeunterbrechungen auftreten werden. Oder Ihr Kunde kann die API im Laufe des Tages viele Male aufrufen, um den Abruf und die Nutzung zu definieren. Berücksichtigen Sie bei der Auswahl der API-Aufrufhäufigkeit die [maximale Anzahl aktiver Prefetch-Zeitpläne](#) und die Wahrscheinlichkeit, dass sich Ihr Zeitplan für Werbeunterbrechungen ändert, nachdem Sie Ihre Prefetch-Zeitpläne erstellt haben. Wenn es wahrscheinlich ist, dass sich der Zeitplan für Werbeunterbrechungen ändert, nachdem Sie Ihre Prefetch-Zeitpläne erstellt haben, sollten Sie die API möglicherweise häufiger aufrufen.

Wiederkehrender Prefetch-Zeitplan

Um einen Zeitplan zu erstellen, bei dem Anzeigen vorab abgerufen werden, bevor jede Anzeige in einem Event verfügbar ist.

1. Geben Sie im Bereich Wiederkehrender Abruf die Abrufeinstellungen an, die Sie verwenden möchten. Diese Einstellungen bestimmen, wann Anzeigen aus dem ADS MediaTailor vorab

abgerufen werden. Sie legen auch fest, welche dynamischen Sitzungsvariablen gegebenenfalls in die Anfrage an das ADS aufgenommen werden sollen.

- Geben Sie unter Wiederkehrendes Prefetch-Fenster die Uhrzeit ein, zu der Prefetch-Abrufe für diese Werbepause gestartet werden MediaTailor können. MediaTailor versucht, Anzeigen für Manifestanfragen, die Ihr Kunde an oder nach diesem Zeitpunkt gestellt hat, vorab abzurufen. Der Standardwert ist die aktuelle Zeit. Wenn Sie keinen Wert angeben, beginnt der Dienst so bald wie möglich mit dem Prefetch-Abruf.
 - Geben Sie unter Verzögerung nach Ende der Verfügbarkeit die Anzahl der Sekunden ein, die nach Ablauf einer Verfügbarkeit warten MediaTailor sollen, bevor Anzeigen für die nächste Verfügbarkeit vorab abgerufen werden. Wenn Sie keinen Wert angeben, wird MediaTailor standardmäßig „Keine Verzögerung“ verwendet.
 - Geben Sie für die Dauer des Traffic Shaping-Fensters die Anzahl der Sekunden ein, über die Anfragen an das ADS verteilt werden MediaTailor sollen. Weitere Informationen finden Sie unter Erläuterung des [Zeitplans für wiederkehrende Prefetch-Abrufe](#)
 - Geben Sie im Abschnitt [Dynamische Variablen](#) bis zu 100 dynamische Sitzungsvariablen ein. MediaTailor verwendet diese Variablen als Ersatz in Prefetch-Anfragen, die es an das ADS sendet. [Wenn Sie keine dynamischen Sitzungsvariablen eingeben, versucht es nach besten MediaTailor Kräften, die Werte für die in Ihrer ADS-URL enthaltenen dynamischen Variablen zu interpolieren.](#)
 - Wählen Sie Dynamische Variable hinzufügen aus.
 - Geben Sie als Schlüssel einen dynamischen Sitzungsvariablenschlüssel ein, z. `scte.event_id` B. Sie können jede dynamische Variable verwenden, die MediaTailor unterstützt. Hinweise zu dynamischen Sitzungsvariablen finden Sie unter [Verwenden von Sitzungsvariablen](#).
 - Geben Sie unter Wert einen dynamischen Variablenwert ein, z. *my-event* B.
 - Um eine weitere dynamische Variable hinzuzufügen, wählen Sie „Dynamische Variable hinzufügen“.
2. Geben Sie im Bereich Verbrauch die Einstellungen an, die Sie für das Verbrauchsfenster verwenden möchten. Diese Einstellungen bestimmen, wann MediaTailor die Anzeigen in den Werbeblock aufgenommen werden. Sie bestimmen auch alle verfügbaren Übereinstimmungskriterien, die Sie verwenden möchten.
- Geben Sie unter Ablauf der abgerufenen Anzeige an, wie lange nach dem Abruf Anzeigen zum Einfügen verfügbar sind.

- Wählen Sie im Abschnitt [Übereinstimmungskriterien verfügbar](#) die Option Verfügbarkeitskriterien hinzufügen aus und fügen Sie Ihrem Zeitplan bis zu fünf verfügbare Kriterien hinzu. Fügen Sie dann unter Dynamischer Variablenschlüssel einen dynamischen Variablenschlüssel hinzu, z. B. `scte.event_id`. MediaTailor platziert die vorab abgerufenen Anzeigen nur dann im Werblock, wenn sie die Kriterien erfüllen, die durch die dynamischen Variablenwerte definiert sind. MediaTailor, an die Ihr Kunde entweder weiterleitet oder die MediaTailor aus Informationen wie Sitzungsdaten abgeleitet werden. Wenn eine Werbeunterbrechung die angegebenen Übereinstimmungskriterien nicht erfüllt, wird der Vorabruf für diese Werbepause MediaTailor übersprungen. Weitere Informationen finden Sie in der Erläuterung der Nutzung des Zeitplans für [wiederkehrende Prefetches](#).

3. Wählen Sie „Verfügbarkeitskriterien hinzufügen“ aus.

Prefetch-Zeitpläne laufen automatisch ab, wenn das Nutzungsfenster abgelaufen ist. Zu Diagnosezwecken bleiben sie mindestens 7 Tage lang sichtbar und MediaTailor werden danach automatisch gelöscht. Alternativ können Sie einen Prefetch-Zeitplan jederzeit manuell löschen. Informationen zum manuellen Löschen eines Prefetch-Zeitplans finden Sie im folgenden Abschnitt. [the section called “Löschen von Prefetch-Zeitplänen”](#)

Löschen von Prefetch-Zeitplänen

Im folgenden Verfahren wird erklärt, wie Sie einen Prefetch-Zeitplan mithilfe der Konsole löschen. MediaTailor Informationen zum programmgesteuerten Löschen von Prefetch-Zeitplänen mithilfe der MediaTailor API finden Sie in der API-Referenz. [DeletePrefetchSchedule](#) AWS Elemental MediaTailor

Note

Das Löschen erfolgt nicht in Echtzeit. Beim MediaTailor Löschen der Prefetch-Zeitpläne kann es zu Verzögerungen kommen. Während dieser Zeit laufen der Prefetch-Abruf und die Nutzung weiterhin im Hintergrund.

Um einen Prefetch-Zeitplan mit der Konsole zu löschen

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>
2. Wählen Sie im Navigationsbereich Konfigurationen aus. Wählen Sie die Wiedergabekonfiguration aus, die die Prefetch-Zeitpläne enthält, die Sie löschen möchten.

3. Wählen Sie auf der Registerkarte Prefetch-Zeitpläne den Prefetch-Zeitplan aus, den Sie löschen möchten. Wählen Sie dann Löschen aus.

Verwenden von vorkonditionierten Anzeigen mit AWS Elemental MediaTailor

In einem [typischen Arbeitsablauf zum Einfügen von Anzeigen](#) werden Anzeigen MediaTailor dynamisch transkodiert, sodass sie dem Inhaltsstream entsprechen, gespeichert und in den Livestream integriert. Da dieser Vorgang erst stattfindet, nachdem die Anzeige in einer VAST-Antwort vom Ad Decision Server (ADS) MediaTailor empfangen wurde, kommt es zu einer Verzögerung, bis die Anzeige zum Zusammenfügen verfügbar ist. Wenn beim Ad Stitching-Workflow zusätzliche Latenz entsteht (entweder aufgrund eines ADS-Timeouts oder anderer Inhalts- oder Netzwerkprobleme), MediaTailor kann dies dazu führen, dass die Verfügbarkeit teilweise aufgefüllt wird oder die Werbeunterbrechung ganz verpasst wird.

Um den Zeitaufwand für das Einfügen von Anzeigen in Ihre Inhalte zu reduzieren, können Sie vorkonditionierte Anzeigen verwenden. Eine vorkonditionierte Anzeige ist eine Anzeige, die Sie transkodieren, bevor Sie sie bei der Anzeigeneinfügung verwenden. MediaTailor Anstatt die unkonditionierten Anzeigen URLs für Ihre ADS bereitzustellen, stellen Sie die URLs für die vorkonditionierten Anzeigen bereit. In seiner VAST-Antwort auf die MediaTailor Anfrage enthält das ADS direkte Links zu den vorkonditionierten Anzeigen. Durch das Entfernen des transkodierenden Teils des Anzeigen-Stitchings MediaTailor muss die Anzeige lediglich gespeichert und in den Inhaltsstream eingefügt werden. Beim Anzeigen-Stitching-Verfahren mit vorkonditionierten Anzeigen wird die Zeit zwischen dem Zeitpunkt, an dem die Anzeige durch die VAST-Antwort auf eine Anzeige aufmerksam gemacht MediaTailor wird, und dem Zeitpunkt, an dem die Anzeige in den Inhalt eingebunden wird, verkürzt.

Alternativ können Sie auch das Ad Prefetching verwenden. In diesem Fall konfigurieren Sie, dass der Ad Stitching-Vorgang MediaTailor zu einem festgelegten Zeitpunkt ausgeführt wird, bevor die Werbepause erforderlich ist. Weitere Informationen zum Vorabruf von Anzeigen finden Sie unter [Anzeigen vorab abrufen](#)

Anforderungen an vorkonditionierte Anzeigen

Die folgenden Anforderungen müssen bei der Einrichtung eines Workflows zur Anzeigenzusammenstellung mit vorkonditionierten Anzeigen berücksichtigt werden.

MediaFiles-Voraussetzungen

Die VAST-Antwort, an die der Ad-Server sendet, MediaTailor muss enthalten MediaFiles, dass diese Anforderungen erfüllt sind:

Die Anzeige (Creative) muss Varianten enthalten, die den Bitratenvarianten des Inhaltsstreams entsprechen. Es liegt in Ihrer Verantwortung, sicherzustellen, dass die VAST-Antwort die richtigen Anzeigenvarianten verwendet, die mit den Vorlagenmanifesten übereinstimmen.

Die Verwendung von vorkonditionierten Anzeigen kann zwar dazu beitragen, die Anzeigeneinblendung effizienter zu gestalten, ist MediaTailor jedoch nicht in der Lage, den Transcodierungsprozess zu verwalten, um sicherzustellen, dass die Mediendateien für die Anzeigen mit den Spezifikationen der Inhaltsmanifeste kompatibel sind. Wenn die Anzeige nicht mit dem Content-Stream übereinstimmt, MediaTailor könnte die Einblendung verpasst werden, oder die Nichtübereinstimmung könnte zu einem Fehler auf dem Wiedergabegerät führen.

Um ohne MediaTailor Transcodierung in den Inhaltsstream eingebunden zu werden, MediaFile muss eine außerdem die folgenden Anforderungen erfüllen:

- Es muss im öffentlichen Internet zugänglich sein, damit es heruntergeladen MediaTailor werden kann.
- Es muss die Streaming-Bereitstellung verwenden, wie `delivery="streaming"` in der VAST-Antwort angegeben.
- Es muss sich entweder um eine `.m3u8` (für HLS) oder `.mpd` (für DASH) Datei handeln.

Example RIESIGE Antwort

Aus der folgenden VAST-Beispielantwort MediaTailor MediaFile fügt Folgendes ein URLs:

- Für einen HLS-Stream MediaTailor verwendet `https://example-ad-origin.amazonaws.com/ad1/index_low.m3u8`. Dies ist die erste Version MediaFile mit Streaming-Bereitstellung und unterstützter Dateierweiterung (`.m3u8`).
- Für einen DASH-Stream MediaTailor verwendet `https://example-ad-origin.amazonaws.com/ad1/index.mpd`. Dies ist der erste MediaFile mit Streaming-Bereitstellung und einer unterstützten Dateierweiterung (`.mpd`).

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="3.0">
  <Ad id="ad1">
    <InLine>
      <AdSystem>ExampleAdSystem</AdSystem>
      <AdTitle>ad1</AdTitle>
      <Impression><![CDATA[https://example-impression.amazonaws.com]]></
Impression>
      <AdServingId>de8e0d33-9c72-4d77-bb3a-f7e566ffc605</AdServingId>
      <Creatives>
        <Creative id="creativeId1" sequence="1">
          <Linear skipoffset="00:00:05">
            <Duration>00:00:30</Duration>
            <MediaFiles>
              <MediaFile delivery="progressive" width="1280" height="720"
type="video/mp4" bitrate="533" scalable="true" maintainAspectRatio="true"><![
CDATA[https://example-ad-origin.amazonaws.com/ad1/ad1.mp4]]></MediaFile>
              <MediaFile delivery="streaming" width="1280"
height="720" type="application/dash+xml" bitrate="533" scalable="true"
maintainAspectRatio="true"><![CDATA[https://example-ad-origin.amazonaws.com/ad1/
index.mpd]]></MediaFile>
              <MediaFile delivery="streaming" width="640"
height="360" type="application/x-mpegURL" bitrate="262" scalable="true"
maintainAspectRatio="true"><![CDATA[https://example-ad-origin.amazonaws.com/ad1/
index_low.m3u8]]></MediaFile>
              <MediaFile delivery="streaming" width="2560"
height="1440" type="application/x-mpegURL" bitrate="1066" scalable="true"
maintainAspectRatio="true"><![CDATA[https://example-ad-origin.amazonaws.com/ad1/
index_high.m3u8]]></MediaFile>
            </MediaFiles>
          </Linear>
        </Creative>
      </Creatives>
    </InLine>
  </Ad>
</VAST>

```

Und offensichtliche Anforderungen

Um vorkonditionierte Anzeigen verwenden zu können, müssen die Anzeigenlisten für Eltern und Kinder die folgenden Anforderungen erfüllen:

- Bei dem Manifest, auf das im Creative Abschnitt der VAST-Antwort verwiesen wird, muss es sich um das übergeordnete Anzeigenmanifest handeln.

- URLs Bei den Anzeigenmanifesten für das untergeordnete Objekt muss es sich um relative Pfade handeln.
- Die untergeordneten Anzeigenmanifeste müssen sich im selben Verzeichnis wie das übergeordnete Manifest befinden, also auf derselben Ebene. Untergeordnete Manifeste dürfen sich nicht in einem Unterverzeichnis oder an einem anderen Ort befinden.

Example unterstütztes übergeordnetes Manifest

Das folgende übergeordnete Anzeigenmanifest enthält relative Anzeigenmanifeste URLs für untergeordnete Anzeigen. Die untergeordneten Manifeste befinden sich ebenfalls im selben Verzeichnis wie das übergeordnete Manifest.

```
#EXTM3U
#EXT-X-STREAM-INF:BANDWIDTH=150000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
index_1.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=440000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
index_2.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=640000,RESOLUTION=640x360,CODECS="avc1.42e00a,mp4a.40.2"
index_3.m3u8
```

Example übergeordnetes Manifest wird nicht unterstützt: Unterverzeichnisse

Das folgende übergeordnete Anzeigenmanifest enthält untergeordnete Manifeste, die sich relativ zum übergeordneten Manifest in Unterverzeichnissen befinden. Es ist kein unterstütztes Manifest für vorkonditionierte Anzeigen.

```
#EXTM3U
#EXT-X-STREAM-INF:BANDWIDTH=150000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
child/index_1.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=440000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
child/index_2.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=640000,RESOLUTION=640x360,CODECS="avc1.42e00a,mp4a.40.2"
child/index_3.m3u8
```

Example übergeordnetes Manifest wird nicht unterstützt: absolut URLs

Das folgende übergeordnete Anzeigenmanifest enthält untergeordnete Manifeste mit absoluter URLs Angabe. Es ist kein unterstütztes Manifest für vorkonditionierte Anzeigen.

```
#EXTM3U
```

```
#EXT-X-STREAM-INF:BANDWIDTH=150000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
https://example.mediatailor.us-west-2.amazonaws.com/index_1.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=440000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
https://example.mediatailor.us-west-2.amazonaws.com/index_2.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=640000,RESOLUTION=640x360,CODECS="avc1.42e00a,mp4a.40.2"
https://example.mediatailor.us-west-2.amazonaws.com/index_3.m3u8
```

Arbeitsablauf für vorkonditionierte Anzeigen

Im Folgenden finden Sie eine grundlegende Beschreibung der Funktionsweise vorkonditionierter Anzeigen in einem Ad Stitching-Workflow mit MediaTailor. Der erste Teil des Workflows besteht aus Maßnahmen, die Sie ergreifen müssen, um die Verwendung vorkonditionierter Anzeigen einzurichten. Im zweiten Teil wird beschrieben, wie die Anzeigen MediaTailor verarbeitet werden.

Teil 1: Einrichtung von vorkonditionierten Anzeigen

Gehen Sie wie folgt vor, um einen Workflow einzurichten, der vorkonditionierte Anzeigen in verwendet. MediaTailor

1. Verwenden Sie einen Transcoder-Service, z. B. um Ihre Werbemittel in Varianten zu unterteilen. AWS Elemental MediaConvert, die die verschiedenen Bitraten, Auflösungen und Codecs Ihrer Vorlagenmanifeste unterstützen.
2. Stellen Sie Ihrem ADS die URLs für die vortranskodierte Mediendateien zur Verfügung, damit sie in VAST-Antworten verwendet werden können.
3. [Erstellen Sie Ihre Wiedergabekonfiguration](#) in MediaTailor. Um vorkonditionierte Werbung zu verwenden, wählen Sie in der Konfiguration für die Einstellung Konditionierung von Streaming-Mediendateien die Option Keine aus.
4. Fahren Sie mit der Einrichtung Ihrer Inhaltsbereitstellung wie gewohnt fort.

Teil 2: MediaTailor Anzeigenverarbeitung

MediaTailor Das Zusammenfügen von Anzeigen wird wie unter beschrieben abgeschlossen.

[Wie funktioniert das Einfügen von MediaTailor Anzeigen](#) Wenn die ADS eine VAST-Antwort MediaTailor erhält, bestimmt sie anhand der folgenden Logik, welche Maßnahmen für die Anzeigen ergriffen werden sollen. Diese Logik wird durch die Einstellung zur Konditionierung von Streaming-Mediendateien in der Wiedergabekonfiguration vorgegeben.

- Wenn die Konditionierung von Streaming-Mediendateien auf Transkodieren eingestellt ist, werden die Mediendateien bei der progressive Auslieferung MediaTailor transkodiert und in

das Manifest zusammengefügt. Wenn nicht genügend Mediendateien für Werbeanzeigen mit progressive Versand zur Verfügung stehen, werden diese Dateien MediaTailor transkodiert und bei Lieferung verwendet. `streaming`

- Wenn die Konditionierung von Streaming-Mediendateien auf „Keine“ gesetzt ist, werden Anzeigen mit MediaTailor `streaming` Übertragungsmediendateien in das Manifest zusammengefügt, ohne sie zu transkodieren. Wenn nicht genügend Werbeanzeigen mit Mediendateien zur `streaming` Auslieferung vorhanden sind, werden diese Dateien MediaTailor transkodiert und bei der Auslieferung verwendet. `progressive`

Verwendung dynamischer Anzeigenvariablen in MediaTailor

Die AWS Elemental MediaTailor Anfrage an den Ad Decision Server (ADS) enthält Informationen über die aktuelle Anzeigesitzung, anhand derer der ADS die besten Anzeigen für die Antwort auswählen kann. Wenn Sie die ADS-Vorlage in Ihrer MediaTailor Konfiguration konfigurieren, können Sie dynamische Variablen, auch Makros genannt, einbeziehen. Dynamische Variablen sind austauschbare Zeichenketten.

Dynamische Variablen können die folgenden Formen annehmen:

- Statische Werte — Werte, die sich von einer Sitzung zur nächsten nicht ändern. Beispiel: Der Antworttyp, den MediaTailor von ADS erwartet.
- Domain-Variablen — Dynamische Variablen, die für URL-Domains verwendet werden können, wie z. B. der `my-ads-server.com`-Teil der URL `http://my-ads-server.com`. Details hierzu finden Sie unter [Verwendung von Domänenvariablen](#).
- Sitzungsdaten — Dynamische Werte, die von MediaTailor für jede Sitzung bereitgestellt werden, z. B. die Sitzungs-ID. Details hierzu finden Sie unter [Verwenden von Sitzungsvariablen](#).
- Spielerdaten — Dynamische Werte, die vom Spieler für jede Sitzung bereitgestellt werden. Diese beschreiben den Inhaltsbetrachter und helfen dem ADS dabei, zu bestimmen, welche Anzeigen in den Stream eingefügt werden MediaTailor sollen. Details hierzu finden Sie unter [Verwendung von Spielervariablen](#).

Übergabe von Parametern an das ADS

In den folgenden Schritten wird beschrieben, wie dynamische Variablen in MediaTailor Anfragen an das ADS eingerichtet werden.

- Hinweise zur unterstützten Formatierung von Abfrageparametern finden Sie unter [Unterstützte Zeichen und Einschränkungen des Manifest-Abfrageparameters](#) und [Längenbeschränkungen für ADS-Abfrageparameter](#).
- Weitere Anpassungen der ADS-Anfrage finden Sie unter [Fortgeschrittene Nutzung](#).

So übergeben Sie Sitzungs- und Player-Informationen an den ADS

1. Ermitteln Sie gemeinsam mit dem ADS, welche Informationen benötigt werden, damit es auf eine Anzeigenanfrage von AWS Elemental MediaTailor antworten kann.
2. Erstellen Sie eine Konfiguration MediaTailor , die eine ADS-Vorlagen-URL verwendet, die die ADS-Anforderungen erfüllt. Schließen Sie in die URL statische Parameter und Platzhalter für dynamische Parameter ein. Geben Sie Ihre Vorlagen-URL in das Feld Ad decision server (Entscheidungsserver für Werbung) ein.

In der folgenden Beispielvorlagen-URL bietet `correlation` Sitzungsdaten und `deviceType` bietet Player-Daten:

```
https://my.ads.server.com/path?  
correlation=[session.id]&deviceType=[player_params.deviceType]
```

3. Konfigurieren Sie auf dem Player die Sitzungsinitiiierungs-Anforderung für AWS Elemental MediaTailor , um Parameter für die Player-Daten bereitzustellen. Sie brauchen die Parameter nur in die Sitzungsinitiiierungs-Anforderung einzuschließen und können sie in nachfolgenden Anforderungen für die Sitzung weglassen.

Die Art des Aufrufs, den der Player zur Initialisierung der Sitzung tätigt, bestimmt, ob der Player (Client) oder MediaTailor (Server) Werbeverfolgungsberichte für die Sitzung bereitstellt. Weitere Informationen zu diesen beiden Optionen finden Sie unter [Daten zur Berichterstattung und Nachverfolgung](#) .

Führen Sie je nachdem, ob Sie an server- oder clientseitigen Werbenachverfolgungs-Berichten interessiert sind, den jeweiligen der folgenden Aufrufe durch. Bei beiden Beispielaufrufen ist `userID` für den ADS und `auth_token` für den Ursprungs-Server vorgesehen:

- (Option) Aufruf für serverseitiges Ad-Tracking-Reporting — Stellen Sie den Parametern, die Sie an das ADS senden MediaTailor möchten, ein Präfix voran. `ads` Lassen Sie das Präfix bei Parametern weg, die MediaTailor an den Ursprungs-Server senden soll:

Die folgenden Beispiele zeigen eingehende Anfragen für HLS und DASH an. AWS Elemental MediaTailor verwendet das `deviceType` in seiner Anfrage an das ADS und das `auth_token` in seiner Anfrage an den Ursprungsserver.

HLS-Beispiel:

```
GET master.m3u8?ads.deviceType=ipad&auth_token=kjhdsaf7gh
```

DASH-Beispiel:

```
GET manifest.mpd?ads.deviceType=ipad&auth_token=kjhdsaf7gh
```

- (Option) Aufruf für clientseitiges Ad-Tracking-Reporting — Geben Sie Parameter für das ADS innerhalb eines Objekts an. `adsParams`

HLS-Beispiel:

```
POST master.m3u8
{
  "adsParams": {
    "deviceType": "ipad"
  }
}
```

DASH-Beispiel:

```
POST manifest.mpd
{
  "adsParams": {
    "deviceType": "ipad"
  }
}
```

Wenn der Player eine Sitzung initiiert, werden die Variablen in der ADS-Anforderungs-URL der Vorlage durch die Sitzungsdaten und die Parameter des Players AWS Elemental MediaTailor ersetzt. Die verbleibenden Parameter werden vom Player an den Ursprungs-Server übergeben.

Example MediaTailor Anfragen mit Anzeigenvariablen

Die folgenden Beispiele zeigen die Aufrufe von AWS Elemental MediaTailor an den ADS und den Ursprungs-Server entsprechend den voranstehenden Beispielen für Sitzungsinitialisierungs-Aufrufe des Players:

- MediaTailor ruft das ADS mit Sitzungsdaten und dem Gerätetyp des Players auf:

```
https://my.ads.server.com/path?correlation=896976764&deviceType=ipad
```

- MediaTailor ruft den Ursprungsserver mit dem Autorisierungstoken des Spielers auf.
 - HLS-Beispiel:

```
https://my.origin.server.com/master.m3u8?auth_token=kjhdsaf7gh
```

- DASH-Beispiel:

```
https://my.origin.server.com/manifest.mpd?auth_token=kjhdsaf7gh
```

Unterstützte Zeichen und Einschränkungen des Manifest-Abfrageparameters

Sie können die folgenden Zeichen in Abfrageparametern verwenden, die in Manifestanforderungen verwendet werden:

- Alphanumerisch (A-Z, a-z, 0-9)
- Perioden (.)
- Bindestriche (-)
- Unterstriche (_)
- Schrägstriche (\)

Längenbeschränkungen

Die Gesamtlänge aller Manifest-Abfrageparameter (Schlüssel und Wert zusammen) darf 2000 Zeichen nicht überschreiten.

Nicht unterstützte Zeichen

Sie können die folgenden Zeichen nicht in Manifest-Abfrageparametern verwenden: : ? & = % / (Schrägstrich)

Längenbeschränkungen für ADS-Abfrageparameter

Die folgenden Längenbeschränkungen gelten für Abfrageparameter, die in Anfragen an das ADS verwendet werden:

- ADS-Parametername: 10000 Zeichen
- ADS-Parameterwert: 25000 Zeichen
- ADS-URL: 25000 Zeichen

Fortgeschrittene Nutzung

Sie können die ADS-Anforderung in vielerlei Hinsicht mit Player- und Sitzungsdaten anpassen. Die einzige Voraussetzung ist die Angabe des ADS-Hostnamens.

Die folgenden Beispiele zeigen einige der Möglichkeiten, wie Sie Ihre Anforderung anpassen können:

- Erstellen Sie durch Verkettung von Player-Parametern und Sitzungsparametern neue Parameter. Beispiel:

```
https://my.ads.com?key1=[player_params.value1][session.id]
```

- Verwenden Sie einen Player-Parameter als Teil eines Pfadelements. Beispiel:

```
https://my.ads.com/[player_params.path]?key=value
```

- Verwenden Sie Player-Parameter, um nicht nur die Werte, sondern sowohl Pfadelemente als auch die Schlüssel selbst zu übergeben. Beispiel:

```
https://my.ads.com/[player_params.path]?[player_params.key1]=[player_params.value1]
```

Weitere Informationen zur Verwendung dynamischer Domänen-, Sitzungs- und Player-Variablen finden Sie unter dem entsprechenden Thema.

Themen

- [Verwenden von Domänenvariablen zur Konfiguration mehrerer Inhalts- und Anzeigenquellen](#)

- [Verwenden von Sitzungsvariablen](#)
- [Verwendung von Spielervariablen](#)

Verwenden von Domänenvariablen zur Konfiguration mehrerer Inhalts- und Anzeigenquellen

Mit dynamischen Domänenvariablen können Sie mehrere Domains, wie z. B. den my-ads-server.com-Teil der URL `http://my-ads-server.com`, mit den Player-Parametern in Ihrer Konfiguration verwenden. Auf diese Weise können Sie mehr als eine Inhaltsquelle oder einen Ad Decision Server (ADS) in einer einzigen Konfiguration verwenden.

Sie können Domänenvariablen mit jedem Parameter verwenden, der einen URI enthält:

- `AdDecisionServerUrl`
- `AdSegmentUrlPrefix`
- `ContentSegmentUrlPrefix`
- `LivePreroll.AdDecisionServerUrl`
- `VideoContentSourceUrl`

Domänenvariablen werden zusammen mit Konfigurationsaliasnamen verwendet, um dynamische Variablen zu ersetzen. Konfigurationsalias ordnen den Player-Parametern, die für die dynamische Domänenkonfiguration verwendet werden, eine Reihe von Aliasnamen und Werten zu.

Themen

- [Konfigurationsalias zur Verwendung als dynamische Variablen erstellen](#)
- [Verwendung von Konfigurationsaliasnamen zur dynamischen Konfiguration von Domänen für eine Sitzung](#)

Konfigurationsalias zur Verwendung als dynamische Variablen erstellen

Bevor Sie mit der Verwendung von Domänenvariablen beginnen, erstellen Sie Konfigurationsalias für Ihre Konfiguration. Sie verwenden die Konfigurationsalias bei der Initialisierung der Sitzung als Variablen für den Domänenersatz. Sie können beispielsweise Konfigurationsalias verwenden, um eine Quell-URL während der Sitzungsinitialisierung dynamisch zu konfigurieren.

Konfigurations-Aliase erstellen

Gehen Sie wie folgt vor, um Konfigurationsalias für den Domainersatz mithilfe der MediaTailor Konsole zu erstellen.

So erstellen Sie Konfigurations-Aliase mithilfe der Konsole

1. Öffnen Sie die MediaTailor Konsole unter. <https://console.aws.amazon.com/mediatailor/>
2. Wählen Sie auf der Konfigurationsseite im Abschnitt Konfigurationsalias die Option Player-Parameter hinzufügen aus.
3. Geben Sie den Namen des Player-Parameters ein, den Sie als dynamische Variable für den Domainersatz verwenden möchten. Sie müssen dem Namen ein Präfix voranstellen `player_params..`
4. Wählen Sie OK aus.

AWS Elemental MediaTailor zeigt den neuen Parameter in der Tabelle im Abschnitt Konfigurationsalias an.

5. Jetzt fügen Sie einen Alias und einen Wert hinzu. Wählen Sie den Player-Parameter aus, den Sie gerade benannt haben. Dadurch wird der Abschnitt unter dem Parameternamen erweitert.

Wählen Sie Neuen Alias hinzufügen aus.

6. Geben Sie einen Aliasschlüssel und einen Wert ein. MediaTailor verwendet Value als Ersatzwert für die Domänenvariable.

Verwendung von Konfigurationsaliasnamen zur dynamischen Konfiguration von Domänen für eine Sitzung

Nachdem Sie die Konfigurationsalias eingerichtet haben, können Sie sie als Ersatzvariablen für Domänen in Ihrer Anfrage zur Sitzungsinitialisierung verwenden. Auf diese Weise können Sie die Domänen für Ihre Sitzung dynamisch konfigurieren.

Einschränkungen

Beachten Sie bei der Verwendung von Konfigurationsaliasen die folgenden Einschränkungen:

- Alle in der Domain verwendeten dynamischen Variablen müssen als `ConfigurationAliases` dynamische Variable definiert werden.

- Den Player-Parametervariablen muss ein Präfix vorangestellt werden. `player_params`. Beispiel, `player_params.origin_domain`.
- Die Liste der Aliaswerte muss für jeden Spielerparameter vollständig sein.
- Wenn eine Anforderung für einen dynamischen Wert gestellt wird, der in der Domain verwendet wird, und diese Anfrage entweder nicht die dynamische Variable oder einen der vorkonfigurierten Aliase für diese Variable spezifiziert, schlägt die Anfrage mit einem HTTP-Statuscode fehl. 400

Example Beispiel für die Verwendung

Hier ist ein Beispiel für eine Konfiguration, die Konfigurationsalias und dynamische Domänenvariablen enthält. Achten Sie besonders auf die Player-Parametervariablen, z. B. `[player_params.origin_domain]` in den `VideoContentSourceUrl` Parameterdomänen `AdDecisionServerUrl` und.

```
PUT /playbackConfiguration
{
  "Name": "aliasedConfig",
  ...
  "AdDecisionServerUrl": "https://abc.execute-api.us-west-2.amazonaws.com/ads?
sid=[session.id]&ad_type=[player_params.ad_type]",
  "VideoContentSourceUrl": "https://[player_params.origin_domain].mediapackage.
[player_params.region].amazonaws.com/out/v1/[player_params.endpoint_id]",
  ...
  "ConfigurationAliases": {
    "player_params.origin_domain": {
      "pdx": "abc",
      "iad": "xyz"
    },
    "player_params.region": {
      "pdx": "us-west-2",
      "iad": "us-east-1"
    },
    "player_params.endpoint_id": {
      "pdx": "abcd",
      "iad": "wxyz"
    },
    "player_params.ad_type": {
      "customized": "abc12345",
      "default": "defaultAdType"
    },
  },
},
```

```
...
}
```

Erstellen Sie anhand der obigen Konfiguration eine Anfrage zur Initialisierung der Sitzung und geben Sie dabei die Player-Variablen und Aliase an:

```
POST master.m3u8
{
  "playerParams": {
    "origin_domain": "pdx",
    "region": "pdx",
    "endpoint_id": "pdx",
    "ad_type": "customized"
  }
}
```

MediaTailor ersetzt die Aliaszeichenfolgen durch die zugewiesenen Werte in der Konfiguration der Konfigurationsalias.

Die Anfrage an das ADS sieht wie folgt aus:

```
https://abc.execute-api.us-west-2.amazonaws.com/ads?sid=[session.id]&ad_type=abc12345
```

Die Anfrage an die VideoContentSource sieht so aus:

```
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/abcd
```

Verwenden von Sitzungsvariablen

Um das Senden von Sitzungsdaten an den Ad Decision Server (ADS) zu konfigurieren AWS Elemental MediaTailor, geben Sie in der ADS-URL der Vorlage eine oder mehrere der in diesem Abschnitt aufgeführten Variablen an. Sie können einzelne Variablen verwenden und mehrere Variablen zu einem einzigen Wert verketteten. MediaTailor generiert einige Werte und bezieht den Rest aus Quellen wie dem Manifest und der Anfrage des Players zur Initialisierung der Sitzung.

In der folgenden Tabelle werden die Sitzungsdatenvariablen beschrieben, die Sie in der URL-Konfiguration Ihrer Vorlage für ADS-Anfragen verwenden können. Die in der Tabelle aufgeführten Abschnittsnummern entsprechen der Version 2019a der Spezifikation -35 der Society of Cable

Telecommunications Engineers (SCTE), [Digital Program Insertion Cueing Message](#). Weitere Informationen zu Ad Prefetch finden Sie unter [Anzeigen vorab abrufen](#)

Name	Verfügbar für Ad-Prefetch	Abschnitt mit den SCTE-35-Spezifikationen	Beschreibung
[avail.index]	Ja		Eine Zahl, die die Position einer verfügbaren Anzeige in einem Index angibt. MediaTailor Erstellt zu Beginn einer Wiedergabesitzung einen Index aller verfügbaren Werbeanzeigen in einem Manifest und speichert den Index für den Rest der Sitzung. Wenn eine MediaTailor-Anfrage an das ADS gestellt wird, um die Verfügbarkeit aufzufüllen, enthält diese die Indexnummer der Anzeigenverfügbarkeit. Dieser Parameter ermöglicht es dem ADS, die Anzeigenauswahl unter Verwendung von Funktionen wie Wettbewerbsausschluss und Frequenzumgrenzung zu verbessern.
[avail.random]	Ja		Eine Zufallszahl zwischen 0 und 10.000.000.000 als lange Zahl, die für jede Anfrage an das ADS MediaTailor generiert wird. Einige Werbe-Server verwenden diesen Parameter, um Funktionen wie z. B. die Trennung der Werbung von Konkurrenzunternehmen zu ermöglichen.
[scte.archive_allowed_flag]	Ja	10.3.3.1	Ein optionaler boolescher Wert. Wenn dieser Wert 0 ist, gelten für das Segment Aufzeichnungseinschränkungen. Wenn dieser Wert 1 ist, gelten für das Segment keine Aufnahmeeinschränkungen.
[scte.available_num]	Ja	9.7.2.1	Der Wert, der MediaTailor aus dem SCTE-35-Feld analysiert wurde <code>avail_num</code> , als lange Zahl. MediaTailor kann diesen Wert verwenden, um lineare und verfügbare Zahlen zu bezeichnen.

Name	Verfügbar für Ad-Prefetch	Abschnitt mit den SCTE-35-Spezifikationen	Beschreibung
			Der Wert muss eine Ganzzahl sein.
[scte.ava ils_expected]	Ja	9,7.2.1	Ein optionaler langer Wert, der die erwartete Anzahl von Verfügungen innerhalb des aktuellen Ereignisses angibt.
[scte.delivery_not_restricted_flag]	Ja	10.3.3.1	Ein optionaler boolescher Wert. Wenn dieser Wert 0 ist, sind die nächsten fünf Bits reserviert. Wenn dieser Wert 1 ist, nehmen die nächsten fünf Bits die in der SCTE-35-Spezifikation beschriebenen Bedeutungen an.
[scte.device_restrictions]	Ja	10.3.3.1	Ein optionaler Ganzzahlwert, der drei vordefinierte, unabhängige und nicht hierarchische Gerätegruppen signalisiert. Weitere Informationen zu dieser Variablen finden Sie in der Beschreibung segments_expected in der SCTE-35-Spezifikation.
[scte.event_id]	Ja	9.1 und 9.7.2.1	Der Wert, nach dem MediaTailor aus dem SCTE-35-Feld geparsed wurde <code>splICE_event_id</code> , als lange Zahl. MediaTailor verwendet diesen Wert, um lineare Anzeigenverfügbarkeitsnummern zu kennzeichnen oder um Anzeigenserver-Abfragezeichenfolgen, wie z. B. die Positionen von Ad-Pods, aufzufüllen. Der Wert muss eine Ganzzahl sein.
[scte.no_regional_blackout_flag]	Ja	10.3.3.1	Ein optionaler boolescher Wert. Wenn dieser Wert 0 ist, gelten für das Segment regionale Blackout-Einschränkungen. Wenn dieser Wert 1 ist, gelten die regionalen Blackout-Einschränkungen nicht für das Segment.

Name	Verfügbar für Ad-Prefetch	Abschnitt mit den SCTE-35-Spezifikationen	Beschreibung
[scte.segment_num]	Ja	10.3.3.1	Ein optionaler Integer-Wert, der Segmente innerhalb einer Sammlung von Segmenten nummeriert. Weitere Informationen zu dieser Variablen finden Sie in der Beschreibung segment_num in der SCTE-35-Spezifikation.
[scte.segmentation_event_id]	Ja	10.3.3.1	MediaTailor macht diese Variable verfügbar als scte.event_id
[scte.segmentation_type_id]	Ja	10.3.3.1	Ein optionaler 8-Bit-Ganzzahlwert, der den Segmentierungstyp angibt. Weitere Informationen zu dieser Variablen finden Sie in der Beschreibung segmentation_type_id in der SCTE-35-Spezifikation.

Name	Verfügbar für Ad-Prefetch	Abschnitt mit den SCTE-35-Spezifikationen	Beschreibung
[scte.segmentation_upid]	segmentation_upid_type : Yes private_data : Yes	segmentation_upid: 10.3.3.1 Verwaltet eine private UPID: 10.3.3.3	Entspricht dem SCTE-35-Element. <code>segmentation_upid</code> Das <code>segmentation_upid</code> Element enthält <code>segmentation_upid_type</code> und <code>segmentation_upid_length</code>. MediaTailor unterstützt die folgenden <code>segmentation_upid</code> Typen: • ADS-Informationen (0x0E) — Werbeinformationen. Weitere Informationen finden Sie in der Beschreibung <code>segmentation_upid</code> in der SCTE-35-Spezifikation. • Managed Private UPID (0x0C) — Die Managed Private UPID (MPU) -Struktur, wie in der SCTE-35-Spezifikation definiert. MediaTailor unterstützt binäre oder DASH-XML-SCTE-Repräsentationen. Sie können diese Struktur in einem Podbuster-Workflow verwenden. Geben Sie dazu einen 32-Bit-Wert (4 Byte) an und <code>format_identifizieren</code> Sie die folgenden Parameter in das <code>private_data</code> Attribut auf: <pre>ABCD{"assetId": " my_program ", "cueData": {"cueType": " theAdType ", "key": " pb", "value": " 123456"}}</pre> MediaTailor analysiert die Werte aus dem vorherigen JSON und übergibt sie an die <code>scte.segmentation_upid.cueData.value</code> dynamischen Variablen <code>scte.segmentation_</code>

Name	Verfügbar für Ad-Prefetch	Abschnitt mit den SCTE-35-Spezifikationen	Beschreibung
			<code>upid.assetId</code> <code>scte.segmentation_</code> <code>upid.cueData.key</code> , und. Benutzerdefiniert (0x01) — Eine benutzerdefinierte Struktur. Weitere Informationen finden Sie in der Beschreibung <code>segmentation_upid</code> in der SCTE-35-Spezifikation.
[<code>scte.segmentation_upid.assetId</code>]	Ja		Wird in Verbindung mit der Managed Private UPID (0xC) für Podbuster-Workflows verwendet. <code>segmentation_upid_type</code> MediaTailor leitet diesen Wert aus dem <code>assetId</code> Parameter in der JSON-Struktur der MPU ab. <code>private_data</code> Weitere Informationen finden Sie unter Managed Private UPID JSON structure for a podbuster workflow.
[<code>scte.segmentation_upid.cueData.key</code>]	Ja		Wird in Verbindung mit der Managed Private UPID (0xC) <code>segmentation_upid_type</code> für Podbuster-Workflows verwendet. MediaTailor leitet diesen Wert aus dem <code>cueData.key</code> Parameter in der JSON-Struktur der MPU ab. <code>private_data</code> Weitere Informationen finden Sie unter Managed Private UPID JSON structure for a podbuster workflow.

Name	Verfügbar für Ad-Prefetch	Abschnitt mit den SCTE-35-Spezifikationen	Beschreibung
[scte.segmentation_upid.cueData.value]	Ja		Wird in Verbindung mit der Managed Private UPID (0xC) <code>segmentation_upid_type</code> für Podbuster-Workflows verwendet. MediaTailor leitet diesen Wert aus dem <code>cueData.key</code> Parameter in der JSON-Struktur der MPU ab. <code>private_data</code> Weitere Informationen finden Sie unter Managed Private UPID JSON structure for a podbuster workflow. Der Wert kann eine Zeichenfolge sein.
[scte.segments_expected]	Ja	10.3.3.1	Ein optionaler Integer-Wert, der die erwartete Anzahl einzelner Segmente innerhalb einer Sammlung von Segmenten angibt. Weitere Informationen zu dieser Variablen finden Sie in der Beschreibung <code>segments_expected</code> in der SCTE-35-Spezifikation.
[scte.sub_segment_num]	Ja	10.3.3.1	Ein optionaler Ganzzahlwert, der ein bestimmtes Untersegment innerhalb einer Sammlung von Untersegmenten identifiziert. Weitere Informationen zu dieser Variablen finden Sie in der Beschreibung <code>sub_segment_num</code> in der SCTE-35-Spezifikation.
[scte.sub_segments_expected]	Ja	10.3.3.1	Ein optionaler Ganzzahlwert, der die erwartete Anzahl einzelner Untersegmente innerhalb einer Sammlung von Untersegmenten angibt. Weitere Informationen zu dieser Variablen finden Sie in der Beschreibung <code>sub_segments_expected</code> in der SCTE-35-Spezifikation.

Name	Verfügbar für Ad-Prefetch	Abschnitt mit den SCTE-35-Spezifikationen	Beschreibung
[scte.unique_program_id]	Ja	9.7.2.1	Der Integer-Wert, der MediaTailor aus dem SCTE-35-Feld analysiert wurde. <code>splice_insert unique_program_id</code> Der ADS stellt anhand der eindeutigen Programm-ID (UPID) zielgerichtete Werbung auf Programmebene für lineare Live-Streams bereit. Wenn der SCTE-35-Befehl nicht Splice Insert ist, wird dieser Wert auf einen leeren Wert MediaTailor gesetzt. Der Wert muss eine Ganzzahl sein.

Name	Verfügbar für Ad-Prefetch	Abschnitt mit den SCTE-35-Spezifikationen	Beschreibung
[session.avail_duration_ms]	Ja		Die Dauer des Zeitfensters für die Anzeigenverfügbarkeit in Millisekunden. Der Standardwert ist 300.000 ms. AWS Elemental MediaTailor erhält den Wert für die Dauer wie folgt aus dem Eingabemanifest: - Für HLS: MediaTailor Ruft die Dauer aus den Werten #EXT-X-CUE-OUT: DURATION oder aus den Werten im Tag ab. #EXT-X-DATERANGE. Wenn das Eingabemanifest eine Dauer von Null, ungültig oder 0 für die Anzeigenverfügbarkeit in diesen Tags hat, wird der Standardwert MediaTailor verwendet. - Für DASH: MediaTailor Ruft den Wert für die Dauer aus der Eventdauer ab, sofern eine angegeben wurde. Andernfalls wird der Standardwert verwendet. - Für VOD: Wenn ein VOD-Stream einen Pre-Roll-Werbeaufruf auslöst und das Manifest keine SCTE-Nachrichten mit einem Dauerwert enthält, wird MediaTailor keine Dauer für [session.avail_duration_ms] eingegeben, einschließlich des Standardwerts für die Dauer.
[session.avail_duration_secs]	Ja		Die auf die nächste Sekunde gerundete Dauer des Anzeigenverfügbarkeitsslots oder der Anzeigenverfügbarkeit in Sekunden. MediaTailor bestimmt diesen Wert auf die gleiche Weise, wie er ihn bestimmt [session.avail_duration_ms].

Name	Verfügbar für Ad-Prefetch	Abschnitt mit den SCTE-35-Spezifikationen	Beschreibung
[session.client_ip]	Nein		Die Remote-IP-Adresse, von der die MediaTailor Anfrage kam. Wenn der X-forwarded-for -Header festgelegt ist, dann verwendet MediaTailor diesen Wert für die client_ip .
[session.id]	Nein		Eine eindeutige numerische Kennung für die aktuelle Wiedergabesitzung. Alle Anforderungen eines Players für eine Sitzung haben dieselbe ID, sodass sie für ADS-Felder verwendet werden kann, die zum Korrelieren aller Anforderungen in einer einzelnen Ansicht bestimmt sind.
[session.referer]	Nein		Normalerweise die URL der Seite, auf der der Videoplayer gehostet wird. MediaTailor setzt diese Variable auf den Wert des Referer Headers, den der Player in seiner Anfrage verwendet hat MediaTailor. Wenn der Player diesen Header nicht angibt, lässt MediaTailor die Variable [session.referer] leer. Wenn Sie ein Content Delivery Network (CDN) oder einen Proxy vor dem Manifest-Endpunkt verwenden und möchten, dass diese Variable angezeigt wird, stellen Sie hier den richtigen Header des Players als Proxy bereit.
[session.user_agent]	Nein		Der User-Agent -Header, der von der Sitzungsinitialisierungsanfrage des Players MediaTailor empfangen wurde. Wenn vor dem Manifest-Endpunkt ein CDN oder ein Proxy angegeben wird, müssen Sie den korrekten Header vom Player hier als Proxy einfügen.

Name	Verfügbar für Ad-Prefetch	Abschnitt mit den SCTE-35-Spezifikationen	Beschreibung
[session.uuid]	Nein		Alternative zu. [session.id] Hierbei handelt es sich um eine eindeutige Kennung für die aktuelle Wiedergabesitzung, wie z. B. im Folgenden:
e039fd39-09f0-46b2-aca9-9871cc116cde			

Example

Wenn der ADS erfordert, dass zusammen mit der eindeutigen Sitzungskennung ein Abfrageparameter mit dem Namen `deviceSession` übergeben wird, könnte die ADS-Vorlagen-URL in AWS Elemental MediaTailor wie folgt aussehen:

```
https://my.ads.server.com/path?deviceSession=[session.id]
```

AWS Elemental MediaTailor generiert automatisch einen eindeutigen Bezeichner für jeden Stream und gibt den Bezeichner anstelle von `session.id`. Wenn die ID lautet `1234567`, würde die letzte Anfrage, die MediaTailor an das ADS gestellt wird, in etwa so aussehen:

```
https://my.ads.server.com/path?deviceSession=1234567
```

Wenn für ADS mehrere Abfrageparameter übergeben werden müssen, AWS Elemental MediaTailor könnte die ADS-URL in der Vorlage wie folgt aussehen:

```
https://my.ads.server.com/sample?
e=[scte.avails_expected]&f=[scte.segment_num]&g=[scte.segments_expected]&h=[scte.sub_segment_num]
```

Das folgende Beispiel-XML-Fragment für einen DASH-Marker zeigt, wie man es benutzt:

```
<Period start="PT444806.040S" id="123456" duration="PT15.000S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
```



```

<Event duration="1350000">
  <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="180832" tier="4095">
    <scte35:SpliceInsert spliceEventId="1234567890"
spliceEventCancelIndicator="false" outOfNetworkIndicator="true"
spliceImmediateFlag="false" uniqueProgramId="1" availNum="1" availsExpected="1">
      <scte35:Program><scte35:SpliceTime ptsTime="5672624400"/></scte35:Program>
      <scte35:BreakDuration autoReturn="true" duration="1350000"/>
    </scte35:SpliceInsert>
  </scte35:SpliceInfoSection>

```

Das folgende Beispiel-XML-Fragment für einen DASH-Marker zeigt, wie es verwendet wird `scte35:TimeSignal`:

```

<Period start="PT346530.250S" id="123456" duration="PT61.561S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="5310000">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="183003" tier="4095">
        <scte35:TimeSignal>
          <scte35:SpliceTime ptsTime="3442857000"/>
        </scte35:TimeSignal>
        <scte35:SegmentationDescriptor segmentationEventId="1234567"
segmentationEventCancelIndicator="false" segmentationDuration="8100000"
segmentationTypeId="52" segmentNum="0" segmentsExpected="0">
          <scte35:DeliveryRestrictions webDeliveryAllowedFlag="false"
noRegionalBlackoutFlag="false" archiveAllowedFlag="false" deviceRestrictions="3"/>
          <scte35:SegmentationUpid segmentationUpidType="12"
segmentationUpidLength="2">0100</scte35:SegmentationUpid>
        </scte35:SegmentationDescriptor>
      </scte35:SpliceInfoSection>
    </Event>

```

Das folgende Beispiel-XML-Fragment für einen DASH-Marker zeigt, wie es verwendet wird `scte35:Binary`:

```

<Period start="PT444806.040S" id="123456" duration="PT15.000S">
  <EventStream schemeIdUri="urn:scte:scte35:2014:xml+bin" timescale="1">
    <Event presentationTime="1541436240" duration="24" id="29">
      <scte35:Signal xmlns="http://www.scte.org/schemas/35/2016">
        <scte35:Binary>/DAhAAAAAAAAAAP/wEAUAAAHaf+9/fgAg9YDAAAAAAAAA25aoh</Binary>
      </scte35:Signal>
    </Event>
    <Event presentationTime="1541436360" duration="24" id="30">

```

```

    <scte35:Signal xmlns="http://www.scte.org/schemas/35/2016">

    <scte35:Binary>QW5vdGhlciB0ZXN0IHN0cmVuZyBmb3IgaW5jb2RpbmcgdG8gQmFzZTY0IGVuY29kZWQgYm1uYXJ5Lg=
Binary>
    </scte35:Signal>
  </Event>

```

Das folgende Beispiel für ein HLS-Tag zeigt die Verwendung EXT-X-DATERANGE von:

```

#EXT-X-DATERANGE:ID="splice-6FFFFFF0",START-DATE="2014-03-05T11:
15:00Z",PLANNED-DURATION=59.993,SCTE35-OUT=0xFC002F0000000000FF0
00014056FFFFFF000E011622DCAFF000052636200000000000A0008029896F50
0000087000000000

```

Das folgende Beispiel für ein HLS-Tag zeigt die Verwendung von: EXT-X-CUE-OUT

```

#EXT-OATCLS-SCTE35:/DA0AAAAAAAAAAAAABQb+ADAQ6QAeAhxDVUVJQAAA03/PAAEUrEoICAAAAAag
+2UBNAAANvrtoQ==
#EXT-X-ASSET:CAID=0x00000000020FB6501
#EXT-X-CUE-OUT:201.467

```

Das folgende Beispiel für ein HLS-Tag zeigt die Verwendung von: EXT-X-SPLICEPOINT-SCTE35

```

#EXT-X-SPLICEPOINT-SCTE35:/DA9AAAAAAAAAAP/wBQb+uYbZqwAnAiVDVUVJAAAKqX//
AAEjW4AMEU1EU05CMDAXMTMyMjE5M190NAAAmXz5JA==

```

Das folgende Beispiel zeigt, wie die scte35:Binary Dekodierung verwendet wird:

```

{
  "table_id": 252,
  "section_syntax_indicator": false,
  "private_indicator": false,
  "section_length": 33,
  "protocol_version": 0,
  "encrypted_packet": false,
  "encryption_algorithm": 0,
  "pts_adjustment": 0,
  "cw_index": 0,
  "tier": "0xFFF",
  "splice_command_length": 16,
  "splice_command_type": 5,
  "splice_command": {

```

```

    "splice_event_id": 448,
    "splice_event_cancel_indicator": false,
    "out_of_network_indicator": true,
    "program_splice_flag": true,
    "duration_flag": true,
    "splice_immediate_flag": false,
    "utc_splice_time": {
        "time_specified_flag": false,
        "pts_time": null
    },
    "component_count": 0,
    "components": null,
    "break_duration": {
        "auto_return": false,
        "duration": {
            "pts_time": 2160000,
            "wall_clock_seconds": 24.0,
            "wall_clock_time": "00:00:24:000000"
        }
    },
    "unique_program_id": 49152,
    "avail_num": 0,
    "avails_expected": 0,
    "segment_num": 0,
    "segments_expected": 0,
    "sub_segment_num": 0,
    "sub_segments_expected": 0
},
"splice_descriptor_loop_length": 0,
"splice_descriptors": null,
"Scte35Exception": {
    "parse_status": "SCTE-35 cue parsing completed with 0 errors.",
    "error_messages": [],
    "table_id": 252,
    "splice_command_type": 5
}
}

```

Verwendung von Spielervariablen

Um AWS Elemental MediaTailor zu konfigurieren, dass vom Player empfangene Daten an das ADS gesendet werden, geben Sie in der ADS-URL der Vorlage `player_params.<query_parameter_name>` Variablen an. Wenn der Player beispielsweise einen

`user_id` in seiner Anfrage genannten Abfrageparameter an sendet MediaTailor, um diese Daten in der ADS-Anfrage zu übergeben, fügen Sie diese `[player_params.user_id]` in die ADS-URL-Konfiguration ein.

Auf diese Weise können Sie kontrollieren, welche Abfrageparameter in die ADS-Anforderung eingeschlossen werden. In der Regel fügen Sie einen vom ADS erkannten speziellen Abfrageparameter zur ADS-Anforderungs-URL hinzu und geben Schlüssel-Wert-Paare als Wert des Parameters an.

Die Beispiele im folgenden Verfahren verwenden die folgenden Schlüssel-Wert-Paare:

- `param1` mit einem Wert von `value1`:
- `param2` mit einem Wert von `value2`:

So fügen Sie Abfrageparameter als Schlüssel-Wert-Paare hinzu

1. Konfigurieren Sie unter die URL der ADS-Anforderungsvorlage so, dass sie auf die Parameter verweist. AWS Elemental MediaTailor Die folgende URL veranschaulicht den Einschluss der Beispielparameter:

```
https://my.ads.com/path?param1=[player_params.param1]&param2=[player_params.param2]
```

2. (Optional) URL-codieren Sie für die serverseitige Web-Nachverfolgungsberichte die Schlüssel-Wert-Paare auf dem Player. Beim MediaTailor Empfang der Anfrage zur Sitzungsinitialisierung werden die Werte einmal per URL dekodiert, bevor sie in die ADS-Anforderungs-URL eingefügt werden.

 Note

Wenn Ihr ADS einen URL-codierten Wert benötigt, URL-codieren Sie den Wert zweimal auf dem Player. Auf diese Weise MediaTailor führt die Dekodierung von zu einem einmal codierten Wert für das ADS.

Beispiel: Wenn die an den ADS gesendete Darstellung der Werte decodiert `param1=value1:&param2=value2:` ergibt, dann ist die URL-codierte Darstellung `param1=value1%3A&param2=value2%3A`.

3. Übergeben Sie beim Aufruf zur Sitzungsinitialisierung vom Player die Schlüssel-Wert-Paare MediaTailor als Wert eines einzelnen Abfrageparameters an. Die folgenden Beispielaufrufe stellen die Beispiel-Schlüssel-Wert-Paare für die server- und clientseitigen Werbenachverfolgungs-Berichte bereit.

- Beispielanforderungen für serverseitige Nachverfolgungsberichte – mit URL-codierten Paaren

HLS:

```
<master>.m3u8?ads.param1=value1%3A&ads.param2=value2%3A
```

DASH:

```
<manifest>.mpd?ads.param1=value1%3A&ads.param2=value2%3A
```

- Beispielanforderung für clientseitige Werbenachverfolgungs-Berichte – ohne URL-Codierung

HLS:

```
POST <master>.m3u8
{
  "adsParams": {
    "param1": "value1:",
    "param2": "value2:"
  }
}
```

DASH:

```
POST <manifest>.mpd
{
  "adsParams": {
    "param1": "value1:",
    "param2": "value2:"
  }
}
```

Für serverseitige Berichte werden die Parameter MediaTailor dekodiert, wenn die Spieleraanfrage empfangen wird. Bei der clientseitigen Berichterstattung werden die in der JSON-Nutzlast empfangenen Parameter nicht geändert. MediaTailor sendet die folgende Anfrage an das ADS:

```
https://my.ads.com/<path>?param1=value1:&param2=value2:
```

Auf diese Weise werden die Schlüssel-Wert-Paare param1 und param2 als First-Class-Abfrageparameter in die ADS-Anforderung eingeschlossen.

Übergabe von AWS Elemental MediaTailor Sitzungsinitialisierungsparametern an das Manifest

AWS Elemental MediaTailor kann Abfrageparameter aus der Sitzungsinitialisierung beibehalten und sie an die personalisierte Manifest-URL anhängen, die an den Client-Player zurückgegeben wird. Nachfolgende Client-Anfragen enthalten auch die angehängten Abfrageparameter.

Manifest-Abfrageparameter sind nützlich, wenn Sie ein Content Delivery Network (CDN) zwischen MediaTailor und dem Client Player verwenden, wobei das CDN die Abfrageparameter für Folgendes verwendet:

- Dynamisches Routing zu verschiedenen Endpunkten MediaTailor
- Token-Autorisierung

MediaTailor fügt für die clientseitige Berichterstattung Abfrageparameter für clientseitige Berichtsendpunkte an, aber die Abfrageparameter für die (oder andere CDN-) Segmente werden nicht angehängt. CloudFront

Um die Parametererhaltung zu verwenden, [reichen Sie ein AWS Support-Ticket](#) ein, um die Aktivierung der Weitergabe von Manifestabfrageparametern zu beantragen.

Das Verhalten variiert zwischen HLS und DASH sowie bei der expliziten und impliziten Sitzungsinitialisierung. In den folgenden Themen wird beschrieben, wie Anfragen zur Sitzungsinitialisierung so konfiguriert werden, dass Parameter an MediaTailor das Manifest übergeben werden.

Themen

- [Implizite HLS-Sitzungsinitialisierung mit AWS Elemental MediaTailor](#)
- [Implizite DASH-Sitzungsinitialisierung mit AWS Elemental MediaTailor](#)
- [Explizite Initialisierung von HLS- und DASH-Sitzungen mit AWS Elemental MediaTailor](#)

Implizite HLS-Sitzungsinitialisierung mit AWS Elemental MediaTailor

Wenn die Anforderung Abfrageparameter mit dem Schlüssel enthält `manifest.*`, wie im folgenden Beispiel gezeigt, MediaTailor werden die Abfrageparameter in die Links zu Ressourcen aufgenommen. MediaTailor Links enthalten das `manifest.` Präfix nicht.

```
GET /v1/master/111122223333/originId/index.m3u8?manifest.test=123&other=456
```

Example übergeordnetes Manifest

Im folgenden Beispiel sind die Abfrageparameter MediaTailor für die URL für das übergeordnete Manifest MediaTailor enthalten.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-INDEPENDENT-SEGMENTS
#EXT-X-MEDIA:LANGUAGE="eng",AUTOSELECT=YES,FORCED=NO,TYPE=SUBTITLES,URI="../../../../
manifest/111122223333/originId/session/1.m3u8?manifest.test=123",GROUP-
ID="subtitles",DEFAULT=YES,NAME="caption_1"
#EXT-X-STREAM-INF:CODECS="avc1.640029,mp4a.40.2",AVERAGE-
BANDWIDTH=2525657,RESOLUTION=960x540,SUBTITLES="subtitles",FRAME-
RATE=29.97,BANDWIDTH=2665212
../../../../manifest/111122223333/originId/session/0.m3u8?manifest.test=123
```

Example untergeordnetes Manifest

Schließt im folgenden Beispiel MediaTailor die Abfrageparameter in die URLs für die Inhaltssegmente ein.

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:28716269
#EXT-X-DISCONTINUITY-SEQUENCE:0
#EXTINF:6.006,
https://origin.com/contentSegment_1.ts?originQueryParam=foo
#EXT-X-DISCONTINUITY
#EXTINF:6.006,
../../../../segment/111122223333/originId/session/0/2?manifest.test=123
```

Implizite DASH-Sitzungsinitialisierung mit AWS Elemental MediaTailor

Der Client stellt eine Manifestanforderung ohne Sitzung, wie im folgenden Beispiel gezeigt.

```
GET /v1/dash/111122223333/originId/index.mpd?manifest.test=123&other=456
```

MediaTailor erstellt eine Sitzung für den Client und leitet sie mit den Abfrageparametern weiter:

```
/v1/dash/111122223333/originId/index.mpd?sessionId=session&manifest.test=123
```

Wenn der Client die Anfrage stellt, MediaTailor antwortet er mit einem DASH-Manifest, das dem folgenden Beispiel ähnelt. Die erste Periode ist eine Inhaltsperiode, weshalb der Manifest-Abfrageparameter dort MediaTailor nicht eingefügt wird. In der zweiten Periode, bei der es sich um eine Werbeperiode handelt, wird der Manifest-Abfrageparameter in das `initialization` Attribut und `media` das Attribut des `SegmentTemplate` Elements MediaTailor eingefügt. Das `Location` Element hat auch die Manifest-Abfrageparameter.

```
<?xml version="1.0" encoding="UTF-8"?>
<MPD availabilityStartTime="2018-07-27T09:48:23.634000+00:00"
  id="201" minBufferTime="PT30S" minimumUpdatePeriod="PT15S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011" publishTime="2023-02-14T23:37:43"
  suggestedPresentationDelay="PT25.000S" timeShiftBufferDepth="PT56.997S" type="dynamic"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/ittf/
PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd">
  <BaseURL>https://origin.com/contentSegments/</BaseURL>
  <Location>https://mediatailor.com/v1/dash/111122223333/originId/index.mpd?
manifest.test=123&aws.sessionId=session</Location>
  <Period duration="PT29.963S" id="28737823" start="PT143732873.178S">
    <AdaptationSet bitstreamSwitching="true" mimeType="video/mp4"
segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
      <Representation bandwidth="2200000" codecs="avc1.640029"
frameRate="30000/1001" height="540" id="1" width="960">
        <SegmentTemplate initialization="index_video_7_0_init.mp4?
m=1611174111" media="index_video_7_0_${Number$.mp4?m=1611174111"
presentationTimeOffset="4311986195351" startNumber="28737828" timescale="30000">
          <SegmentTimeline>
            <S d="180180" t="4311986911066"/>
            <S d="3003" t="4311987091246"/>
          </SegmentTimeline>
        </Representation>
      </AdaptationSet>
    </Period>
  </MPD>
```



```

        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
</Period>
<Period id="28737829_1" start="PT39925H48M23.141S">
  <BaseURL>https://mediatailor.com/v1/
dashsegment/111122223333/originId/session/28737829/28737829_1/</BaseURL>
  <AdaptationSet bitstreamSwitching="false" frameRate="30000/1001"
  mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
  subsegmentAlignment="true" subsegmentStartsWithSAP="1">
    <SegmentTemplate startNumber="1" timescale="90000"/>
    <Representation bandwidth="2200000" codecs="avc1.64001f" height="540"
    id="1" width="960">
      <SegmentTemplate initialization="asset_540_2_0init.mp4?
manifest.test=123" media="asset_540_2_0_$.Number%09d$.mp4?manifest.test=123"
      startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180180" r="6" t="0"/>
          <S d="87087" t="1261260"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
</Period>
</MPD>

```

Explizite Initialisierung von HLS- und DASH-Sitzungen mit AWS Elemental MediaTailor

Wenn der Client eine explizite Anfrage zur Sitzungsinitialisierung stellt, MediaTailor werden die AS-Abfrageparameter in das manifestParams übergeordnete Manifest und die Nachverfolgung URLs in die Antwort aufgenommen.

Example Anfrage zur Sitzungsinitialisierung

```

POST /v1/session/111122223333/originId/index.m3u8
{
  "adsParams": {
    "param1": "value1",
    "param2": "value2",
    "param3": "value3"
  }
}

```

```
    },  
    "manifestParams": {  
        "test": "123"  
    }  
}
```

Example Manifest und Antwort zur Nachverfolgung

```
{  
  "manifestUrl": "/v1/master/111122223333/originId/index.m3u8?  
aws.sessionId=session&test=123",  
  "trackingUrl": "/v1/tracking/111122223333/originId/session?test=123"  
}
```

Manifestantworten für die Sitzung haben einen spezifischen manifestParams Charakter, der den zuvor beschriebenen impliziten Workflows zur Sitzungsinitialisierung MediaTailor URLs ähnelt. Der Hauptunterschied besteht darin, dass die Manifestparameter für die explizite Sitzungsinitialisierung nicht mit `manifest.` beginnen.

Manifest-Abfrageparameter sind unveränderlich und werden nur bei der Sitzungsinitialisierung festgelegt. Wenn ein Client mehrere übergeordnete Manifestanfragen für eine einzelne Sitzung stellt, werden die Manifest-Abfrageparameter nach der ersten Anfrage MediaTailor nicht aktualisiert.

Daten zur Berichterstattung und Nachverfolgung

MediaTailor bietet zwei Optionen für die Nachverfolgung und Berichterstattung darüber, wie viel von einer Anzeige sich ein Zuschauer angesehen hat. Bei der serverseitigen Anzeigenberichterstattung wird die Anzeige MediaTailor nachverfolgt und Beacons (Tracking-Signale) direkt an den Anzeigenserver gesendet. Alternativ verfolgt beim clientseitigen Tracking-Ansatz der Client-Player (das Gerät des Nutzers) die Anzeige und sendet die Beacons an den Anzeigenserver. Die Art der Anzeigenberichterstattung, die in einer Playback-Sitzung verwendet wird, hängt von der spezifischen Anfrage ab, mit der der Player die Sitzung initiiert. MediaTailor

Themen

- [Serverseitiges Ad-Tracking](#)
- [Kundenseitiges Ad-Tracking](#)

Serverseitiges Ad-Tracking

AWS Elemental MediaTailor verwendet standardmäßig serverseitige Berichterstattung. Wenn der Player bei der serverseitigen Berichterstellung eine Werbe-URL vom Manifest anfordert, meldet der Service die Werbekonsumierung direkt der Werbeverfolgungs-URL. Nachdem der Player eine Wiedergabesitzung mit MediaTailor eingeleitet hat, ist zum Durchführen der serverseitigen Berichterstellung keine weitere Eingabe von Ihnen oder dem Player erforderlich. MediaTailor Sendet bei der Wiedergabe jeder Anzeige Beacons an den Anzeigenserver, um zu melden, wie viel von der Anzeige angesehen wurde. MediaTailor sendet Beacons für den Beginn der Anzeige und für den Verlauf der Anzeige in Quartilen: erstes Quartil, Mittelpunkt, drittes Quartil und Abschluss der Anzeige.

So führen Sie die serverseitige Werbe-Berichterstellung durch

- Initialisieren Sie vom Player aus eine neue MediaTailor Wiedergabe-Sitzung mit einer Anfrage in einem der folgenden Formate gemäß Ihrem Protokoll:
 - Beispiel: HLS-Format

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/<asset-id>?  
ads.<key-value-pairs-for-ads>&<key-value-pairs-for-origin-server>
```

- Beispiel: DASH-Format

```
GET <mediatailorURL>/v1/dash/<hashed-account-id>/<origin-id>/<asset-id>?ads.<key-  
value-pairs-for-ads>&<key-value-pairs-for-origin-server>
```

Die Schlüssel-Wert-Paare sind die dynamischen Targeting-Parameter für die Werbenachverfolgung. Weitere Informationen zum Hinzufügen von Parametern zur Anforderung finden Sie unter [the section called “Verwendung dynamischer Anzeigenvariablen”](#).

AWS Elemental MediaTailor beantwortet die Anfrage mit der Manifest-URL. Das Manifest enthält URLs für die Medienmanifeste. Die Medien-Manifeste enthalten eingebettete Links für Werbesegment-Anforderungen.

 Note

Wenn in MediaTailor einer Tracking-URL ein doppelter Schrägstrich (//) vorkommt, werden die Schrägstriche zu einem (/) zusammengefasst.

Wenn der Player die Wiedergabe über eine Anzeigensegment-URL (/v1/segmentPfad) anfordert, AWS Elemental MediaTailor sendet er das entsprechende Beacon über das Anzeigen-Tracking an den Anzeigenserver. URLs Gleichzeitig gibt der Service eine Umleitung zum tatsächlichen *.ts-Werbesegment aus. Das Anzeigensegment befindet sich entweder in der CloudFront Amazon-Distribution, in der transkodierte Anzeigen MediaTailor gespeichert werden, oder im Content Distribution Network (CDN), in dem Sie die Anzeige zwischengespeichert haben.

Kundenseitiges Ad-Tracking

Mithilfe der AWS Elemental MediaTailor clientseitigen Tracking-API können Sie Player-Steuerelemente während Werbepausen in Streaming-Workflows integrieren. Beim clientseitigen Tracking sendet der Player oder Client Tracking-Ereignisse wie Impressionen und Quartil-Ad-Beaconing an den Ad Decision Server (ADS) und andere Einheiten zur Anzeigenverifizierung. Bei diesen Ereignissen wird sowohl der Gesamtstatus der Werbeunterbrechung als auch die Anzahl der einzelnen Werbezeiträume innerhalb der einzelnen Werbeunterbrechungen erfasst. Weitere Informationen zu Impression and Quartile (ADS) und anderen Einheiten zur Anzeigenverifizierung finden Sie hier. Weitere Informationen zu Impression und Quartil-Ad-Beaconing finden Sie unter. [Clientseitiges Beaconing](#) Weitere Informationen zu ADS und anderen Entitäten zur Anzeigenverifizierung finden Sie unter. [Kundenseitige Ad-Tracking-Integrationen](#)

Das clientseitige Tracking ermöglicht Funktionen wie die folgenden:

- Countdown-Timer für Werbeunterbrechungen — Weitere Informationen finden Sie unter. [Countdown-Timer hinzufügen](#)
- Durchklicken von Werbeanzeigen — Weitere Informationen finden Sie unter. [Und Click-through](#)
- Anzeige von Begleitanzeigen — Weitere Informationen finden Sie unter. [Begleitende Anzeigen](#)
- Überspringbare Anzeigen — Weitere Informationen finden Sie unter. [Überspringbare Anzeigen](#)
- Anzeige von VAST-Symbolen zur Einhaltung der Datenschutzbestimmungen — Weitere Informationen finden Sie unter. [Symbole für Google Warum diese Anzeige \(WTA\)](#)
- Steuerung des Spieler-Scrubbings bei Werbeanzeigen — Weitere Informationen finden Sie unter [Schrubben](#).

Mithilfe der MediaTailor clientseitigen Tracking-API können Sie Metadaten an das Wiedergabegerät senden, die zusätzlich zum clientseitigen Tracking weitere Funktionen ermöglichen:

Themen

- [Aktivierung der clientseitigen Nachverfolgung](#)
- [Serverparameter hinzufügen](#)
- [Abfrageparameter für Origin-Interaktionen](#)
- [Für die Sitzung konfigurierte Funktionen](#)
- [Bewährte Methoden für die clientseitige Nachverfolgung](#)
- [Schema und Eigenschaften für das clientseitige Ad-Tracking](#)
- [Zeitlicher Ablauf von Ad-Tracking-Aktivitäten](#)
- [Player-Steuerung und Funktionen für kundenseitiges Ad-Tracking](#)
- [Clientseitiges Beaconsing](#)
- [Hybridmodus mit serverseitigen Ad Beacons](#)
- [Kundenseitige Ad-Tracking-Integrationen](#)
- [Durch Werbe-Beacons blättern mit GetTracking](#)

Aktivierung der clientseitigen Nachverfolgung

Sie aktivieren das clientseitige Tracking für jede Sitzung. Der Player sendet ein HTTP POST an den Endpunkt des MediaTailor Sitzungsinitialisierungspräfixes der Konfiguration. Optional kann der Player zusätzliche Metadaten senden, die dann verwendet werden können, wenn er Werbeaufrufe tätigt, den Ursprung für ein Manifest aufruft und MediaTailor Funktionen auf Sitzungsebene aufruft oder deaktiviert. MediaTailor

Das folgende Beispiel zeigt die Struktur der JSON-Metadaten:

```
{
  "adsParams": {
    "param1": "value1",
    "param2": "value2",
    "origin_access_token": "abc123",
  },
  # 'adsParams' is case sensitive
  # key is not case sensitive
  # Values can contain spaces. For example, 'value 2'
  # this is an example of a query parameter designated
  # for the origin
}
```

```
"overlayAvails":"on"           # 'overlayAvails' is case sensitive. This is an
example of a feature that is enabled at the session level.
}
```

Verwenden Sie die MediaTailor Konsole oder API, um die URL der ADS-Anforderungsvorlage so zu konfigurieren, dass sie auf diese Parameter verweist. Im folgenden Beispiel `player_params.param1` sind die Player-Parameter für `param1` und `player_params.param2` die Player-Parameter für `param2`.

```
https://my.ads.com/path?param1=[player_params.param1]&param2=[player_params.param2]
```

Serverparameter hinzufügen

Auf der obersten Ebene der JSON-Struktur befindet sich ein JSON-Objekt. `adsParams` In diesem Objekt befinden sich key/value Paare, die in allen Sitzungsanfragen lesen und an den Ad-Server senden MediaTailor können. MediaTailor unterstützt die folgenden Anzeigenserver:

- Google Ad Manager
- SpringServe
- FreeWheel
- Öffentlich

Abfrageparameter für Origin-Interaktionen

Alle reservierten key/value Paare innerhalb der obersten Ebene der JSON-Struktur, wie, und `adsParams` `availSuppression` `overlayAvails`, werden der ursprünglichen Anforderungs-URL nicht in Form von Abfrageparametern hinzugefügt. Jede Sitzungsmanifestanforderung, die an MediaTailor den Ursprung gerichtet wird, enthält diese Abfrageparameter. Der Ursprung ignoriert überflüssige Abfrageparameter. MediaTailor Kann die key/value Paare beispielsweise verwenden, um Zugriffstoken an den Ursprung zu senden.

Für die Sitzung konfigurierte Funktionen

Verwenden Sie die JSON-Struktur für die Sitzungsinitialisierung, um Funktionen wie, und zu aktivieren, zu deaktivieren oder zu überschreiben. MediaTailor `overlayAvails` `availSuppression` `adSignaling` Alle während der Sitzungsinitialisierung übergebenen Funktionskonfigurationen überschreiben die Einstellung auf der Konfigurationsebene. MediaTailor

Note

Die MediaTailor bei der Sitzungsinitialisierung übermittelten Metadaten sind unveränderlich, und zusätzliche Metadaten können für die Dauer der Sitzung nicht hinzugefügt werden. Verwenden Sie SCTE-35-Markierungen, um Daten zu übertragen, die sich während der Sitzung ändern. Weitere Informationen finden Sie unter [Verwenden von Sitzungsvariablen](#).

Example : Durchführung von kundenseitigem Ad-Tracking für HLS

```
POST mediatailorURL/v1/session/hashed-account-id/origin-id/asset-id.m3u8

{
  "adsParams": {
    "deviceType": "ipad"    # This value does not change during the session.
    "uid": "abdgfdyei-2283004-ueu"
  }
}
```

Example : Durchführung von kundenseitigem Ad-Tracking für DASH

```
POST mediatailorURL/v1/session/hashed-account-id/origin-id/asset-id.mpd

{
  "adsParams": {
    "deviceType": "androidmobile",
    "uid": "xjhddli-9189901-uic"
  }
}
```

Eine erfolgreiche Antwort ist ein HTTP 200 mit einem Antworttext. Der Hauptteil enthält ein JSON-Objekt mit einem `manifestUrl` und einem `trackingUrl` Schlüssel. Bei den Werten handelt es sich um relative Werte URLs , die der Player sowohl für die Wiedergabe als auch für die Nachverfolgung von Werbeereignissen verwenden kann.

```
{
  "manifestUrl": "/v1/dashmaster/hashed-account-id/origin-id/asset-id.m3u8?aws.sessionId=session-id",
  "trackingUrl": "/v1/tracking/hashed-account-id/origin-id/session-id"
}
```

Weitere Informationen zum clientseitigen Tracking-Schema finden Sie unter [Schema und Eigenschaften für das clientseitige Ad-Tracking](#)

Bewährte Methoden für die clientseitige Nachverfolgung

In diesem Abschnitt werden die bewährten Methoden für das clientseitige Tracking sowohl für Live- als auch MediaTailor für VOD-Workflows beschrieben.

Live-Workflows

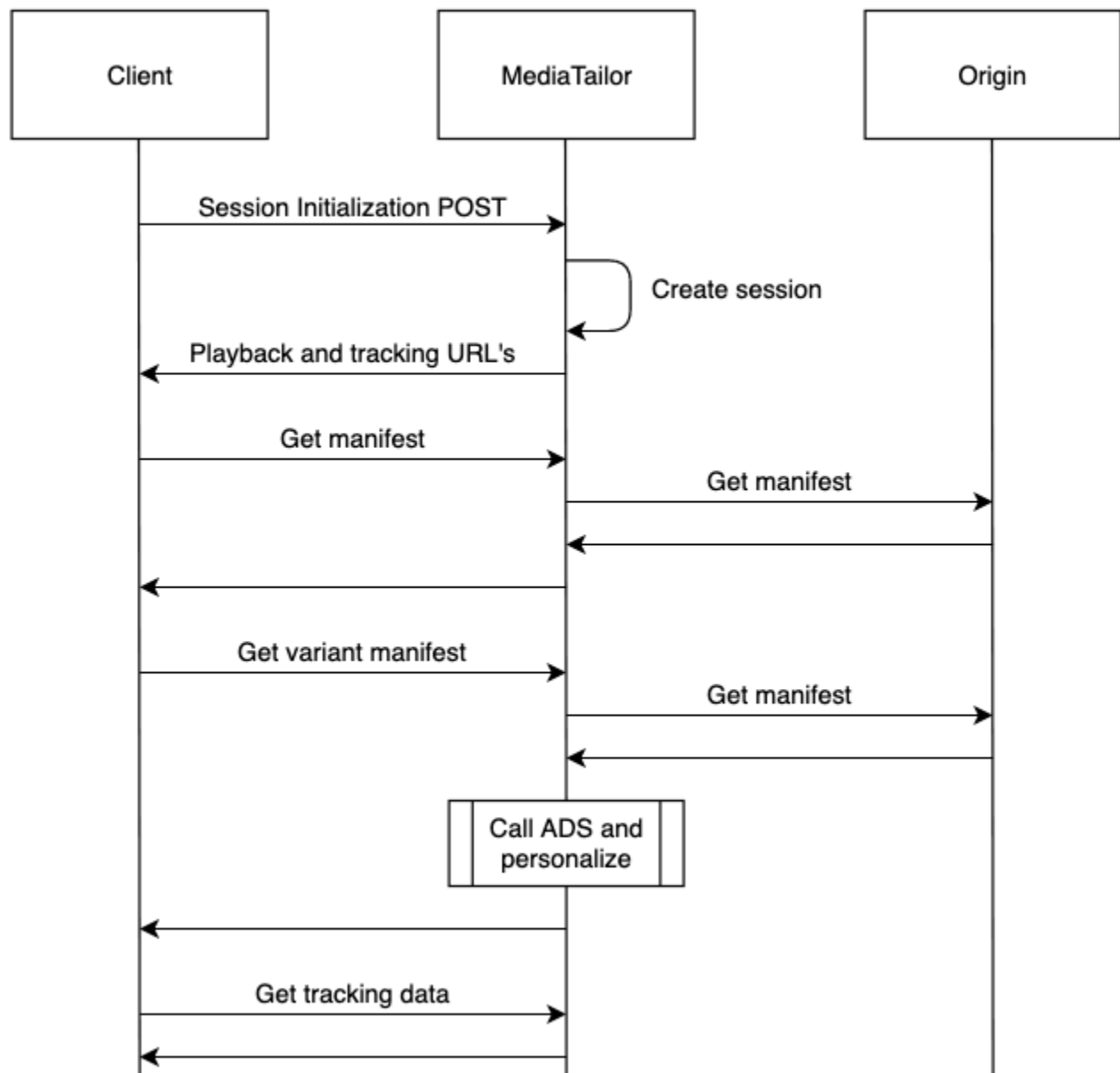
Fragen Sie den Tracking-Endpunkt in einem Intervall ab, das jeder Zieldauer für HLS oder dem minimalen Aktualisierungszeitraum für DASH entspricht, um immer über die aktuellsten Ad-Tracking-Metadaten zu verfügen. Die Anpassung dieses Intervalls ist besonders wichtig in Workflows, bei denen die Kreativen möglicherweise über eine interaktive Komponente oder eine Overlay-Komponente verfügen.

Note

Einige Player unterstützen Event-Listener, die als Alternative zu Umfragen verwendet werden könnten. Beispielsweise müsste die Funktion zur Dekoration von MediaTailor Werbe-IDs für jede Sitzung aktiviert werden. Weitere Informationen finden Sie unter [Werbe-ID-Dekoration](#). Bei Verwendung dieser Funktion wird für jede verfügbare Anzeige ein Datumsbereich (HLS) oder ein Eventelement (DASH) -Identifizier verwendet. Spieler können diese Manifest-Tags als Aufforderung verwenden, um den MediaTailor Tracking-Endpunkt für die Sitzung aufzurufen.

VOD-Workflows

Nach einer erfolgreichen Sitzungsinitialisierung und nach Erhalt des ersten MediaTailor Manifests, das Medien enthält, müssen Sie den Tracking-Endpunkt nur einmal aufrufen.



Schema und Eigenschaften für das clientseitige Ad-Tracking

Mit der MediaTailor clientseitigen Ad-Tracking-Funktion können Sie detaillierte clientseitige Ad-Tracking-Daten in Ihre Player-Umgebung integrieren. In den folgenden Abschnitten werden das allgemeine Ad-Tracking-Schema sowie die spezifischen Eigenschaften und Werte behandelt, aus denen sich das Schema zusammensetzt.

Inhalt

- [Schema](#)

- [Eigenschaften](#)

Schema

In der folgenden Tabelle wird das MediaTailor clientseitige Ad-Tracking-Schema beschrieben. Gegebenenfalls ordnet die Tabelle das Schema VAST-Daten zu.

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
Antwort JSON					
-	Object	avails , nonLinear Avails			
/avails	Array				
MediaTailor erstellt ein Objekt für jeden Avail (Ad-Break) im Manifestfenster.		ads , Anzeigentyp,, availID , durationInSeconds , startTime startTimeInSeconds , dateTime			
/ads	Array				
MediaTailor erstellt für jede Anzeige innerhalb des Verfügbarkeitszeit	Object	adID , AdType,, adParameters , adVerifications , company , Ads , duration , durationInSeconds ,			

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
raums ein Objekt.		Symbole extens s , MediaFile "" dateTime startTi , startTime InSeconds adBreakTr ackingEvents			
/adId	String				• HLS — die Sequenznummer, die dem Anfang der Anzeige zugeordnet ist • DASH — die Perioden-ID der Anzeige

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/adParameters	String		VAST/Ad/Inline/Creatives/Creative/Linear/AdParameters		Zeichenfolge mit Anzeigenparametern aus dem VAST-VPAD, die an MediaTailor den Player weitergegeben wird
/adVerifications	Array	VAST/Ad/Inline/AdVerifications			Enthält die Ressourcen und Metadaten, die für die Ausführung von Messcode eines Drittanbieters zur Überprüfung der kreativen Wiedergabe erforderlich sind

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
MediaTailor erstellt ein Objekt für jedes Element zur Anzeigenv erifizierung.	Object	Ausführbare Ressource n-, Lieferant en- und Überprüfu ngsparame ter javaScrip tResource			
/executableResource	Array		VAST/Ad/I nLine/AdV erificati ons/Verif ication/E xecutable Resource		
MediaTailor erstellt ein Objekt für jedes executableResource Element.	Object	apiFramework , adType, uri, Sprache			

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/apiFramework	String		VAST/Ad/Inline/AdVerifications/Verification/ExecutableResource/@apiFramework		
/type	Zeichenfolge				
/uri	Zeichenfolge		VAST/Ad/Inline/AdVerifications/Verification/ExecutableResource/#CDATA		
/language	String		VAST/Ad/Inline/AdVerifications/Verification/ExecutableResource/@language	VAST/Ad/Inline/AdVerifications/Verification/ExecutableResource/@language	

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/javaScriptResource	Array		VAST/Ad/Inline/AdVerifications/Verification/JavaScriptResource		
MediaTailor erstellt für jedes JavaScriptResource Element ein Objekt.	Object	apiFramework , Browser (optional), uri			
/apiFramework	String		VAST/Ad/Inline/AdVerifications/Verification/JavaScriptResource/@apiFramework		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/browserOptional	Zeichenfolge		VAST/Ad/Inline/AdVerifications/Verification/JavaScriptResource/@browserOptional		
/uri	String		VAST/Ad/Inline/AdVerifications/Verification/JavaScriptResource/#CDATA		
/trackingEvents	Array				
MediaTailor erstellt ein Objekt für jeden Tracking-Ereignistyp eines Elements zur Anzeigenverifizierung.	Object	Ereignis, URI			

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/event	String		VAST/Ad/Inline/AdVerifications/Verification/TrackingEvents/Tracking/@event		
/uri	Zeichenfolge		VAST/Ad/Inline/AdVerifications/Verification/TrackingEvents/Tracking/#CDATA		
/vendor	Zeichenfolge		VAST/Ad/Inline/AdVerifications/Verification/@vendor		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/verificationParameters	String		VAST/Ad/Inline/AdVerification/Verification/VerificationParameters		
/companionAds	Array				Begleitend zeigen, die die Anzeige begleiten, bieten Inhalte wie einen Rahmen um die Anzeige oder ein Banner, das in der Nähe des Videos angezeigt wird.

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
MediaTailor erstellt für jedes Begleitanzeigenelement ein Objekt.	Object	adParameters , altText , attributes , companionClickThrough , companionClickTrackingHTMLResource , Ereignisse verfolgen sequence staticResource	VAST/Ad/Inline/Creatives/Creative/CompanionAds		
/adParameters	String				
/altText	String				
/attributes	Object	adSlotId, apiFramework , AssetHeight, AssetWidth, ExpandHeight, ExpandWidth,, id, pxratio, Rendering Mode, height width			

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/adSlotId	String		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/@adSlotId		
/apiFramework	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/@apiFramework		
/assetHeight	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/@assetHeight		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/assetWidth	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/@assetWidth		
/expandedHeight	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/@expandedHeight		
/expandedWidth	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/@expandedWidth		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/height	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/@height		
/id	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/@id		
/pxratio	Zeichenfolge				
/renderingMode	Zeichenfolge				
/width	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/@width		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/companionClickThrough	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/CompanionClickThrough		
/companionClickTracking	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/CompanionClickTracking		
/htmlResource	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/HTMLResource		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/iFrameResource	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/iFrameResource		
/sequence	Zeichenfolge				
/staticResource	String		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/StaticResource		
/trackingEvents	Array		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/TrackingEvents		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
MediaTailor erstellt ein Objekt für jeden Ereignistyp zur Erfassung von Begleitangeboten.					
/tracking	Object	>Ereignis, URI			
/event	String		VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/TrackingEvents/Tracking/@event		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/uri	Zeichenfolge		VAST/Ad/InLine/Creatives/Creative/CompanionAds/Companion/TrackingEvents/Tracking/#CDATA		
/duration	String				Länge im ISO-8601-Sekunden-Format
/durationInSeconds	Anzahl				Länge im Sekundenformat
/extensions	Array				Werbeserver können benutzerdefinierte VAST-Erweiterungen verwenden

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
MediaTailor erstellt ein Objekt für jede untergeordnete Erweiterung des extensions Elements.			VAST/Ad/Inline/Extensions		
/extension	Object	Typ, Inhalt	VAST/Ad/Inline/Extensions/Extension		
/type	String		VAST/Ad/Inline/Extensions/Extension/@type		
/content	String				
/icons	Array		VAST/Ad/Inline/Creatives/Creative/Linear/Icons		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
MediaTailor erstellt ein Objekt für jedes icon Element darin icons.	Object	attributes , dateTime, duration , durationIn Seconds htmlFrame , iconClick s,, iconViewTracking , iFrameSource , staticResource startTime startTimeInSeconds	VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icons/Icon		
/attributes	Object	apiFramework , durationheight , Offset, Programm, pxratio, XPosition width , YPosition			
/apiFramework	String		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icons/@apiFramework		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/duration	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/@duration		
/height	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/@height		
/offset	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/@offset		
/program	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/@program		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/pxratio	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/@pxratio		
/width	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/@width		
/xPosition	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/@xPosition		
/yPosition	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/@yPosition		
/dateTime	Zeichenfolge				

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/duration	String				
/durationInSeconds	Zahl				
/htmlResource	String		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/HTMLResource		
/iconClicks	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/IconClicks		
/iconClickThrough	String		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/IconClicks/IconClickThrough		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/iconClickTracking	Object	id	VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/IconClicks/IconClickTracking		
/id	String				
/iconClickFallbackImages	Array		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/IconClicks/IconClickFallbackImages		
MediaTailor erstellt ein Objekt für jeden Icon-Click-Fallback-Image-Knoten.					

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/altText	String		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/IconClicks/IconClickFallbackImages/IconClickFallbackImage/AltText		
/height	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/IconClicks/IconClickFallbackImages/IconClickFallbackImage/@height		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/width	String		VAST/Ad/InLine/Creatives/Creative/Linear/Icons/Icon/IconClicks/IconClickFallbackImages/IconClickFallbackImage/@width		
/staticResource	Object	creativeType, uri	VAST/Ad/InLine/Creatives/Creative/Linear/Icons/Icon/IconClicks/IconClickFallbackImages/IconClickFallbackImage/StaticResource		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/creative Type	String		VAST/Ad/InLine/Creatives/Creative/Linear/Icons/Icon/IconClicks/IconClickFallbackImages/IconClickFallbackImage/StaticResource/@creativeType		
/uri	Zeichenfolge		VAST/Ad/InLine/Creatives/Creative/Linear/Icons/Icon/IconClicks/IconClickFallbackImages/IconClickFallbackImage/StaticResource/#CDATA		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/iconViewTracking	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/IconViewTracking		
/iFrameResource	String		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/iFrameResource		
/staticResource	Object	CreativeType, URI	VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/StaticResource		

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/creativeType	String		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/StaticResource/@type		
/uri	Zeichenfolge		VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/StaticResource/#CDATA		
/startTime	String				
/startTimeInSeconds	Zahl				

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/mediaFiles	Object	adParameters , duration , durationInSeconds , mediaFilesListmezzanine , Ereignisse verfolgen startTime , startTimeInSeconds			Video und andere Inhalte, die der Spieler für die Anzeige benötigt, stehen zur Verfügung
/adParameters	String				
/duration	String				
/durationInSeconds	Zahl				
/mediaFilesList	Array				
MediaTailor erstellt ein Objekt für jeden Ereignistyp der Begleitanzeigen-Elementverfolgung		apiFramework , delivery , height , maintainAspectRatio , mediaFileUri , mediaType , scalable , width			

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/apiFrameWork	String				
/delivery	Zeichenfolge				
/height	Zeichenfolge				
/maintainAspectRatio	Zeichenfolge				
/mediaFileUri	Zeichenfolge				
/mediaType	Zeichenfolge				
/scalable	Zeichenfolge				
/width	Zeichenfolge				
/mezzanine	Zeichenfolge				
/startTime	Zeichenfolge				
/startTimeInSeconds	String				
/trackingEvents	Array				

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
MediaTailor erstellt für jedes Tracking-Ereignis ein Objekt für das Werbemittel		beaconUrls , durationInSeconds , dateTime , eventId , eventTime , startTimeInSeconds			
/beaconUrls	Array				
Eine durch Kommas getrennte Liste aller URLs Nachverfolgungen für dieses Ereignis					
/duration	String				
/durationInSeconds	Zahl				
/dateTime	String				
/eventId	Zeichenfolge				

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/eventType	Zeichenfolge				
/startTime	String				
/startTimeInSeconds	Zahl				
/startTime	String				Zeitposition im ISO-8601-Sekunden-Format relativ zum Beginn der Wiedergabe-Sitzung
/startTimeInSeconds	Anzahl				Zeitposition im Sekundenformat relativ zum Beginn der Wiedergabe-Sitzung

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/dateTime	String				Programmdatum und Uhrzeit im Format ISO 8601 Sekunden für den Start der Anzeige verfügbar
/tracking Events	Array				Enthält alle Tracking-Event-Daten, die in der VAST-Antwort empfangen wurden, zusammen mit Zeitinformationen
/adType	String				
/availId	Zeichenfolge				
/dateTime	Zeichenfolge				
/duration	String				
/duration InSeconds	Zahl				

Schlüssel	Datentyp	Schlüssel für Kinder	Zuordnung von VAST 2.0/3.0	Kartierung von VAST 4.0	Beschreibung
/startTime	String				
/startTimeInSeconds	Zahl				

Eigenschaften

In der folgenden Tabelle sind die Eigenschaften in der clientseitigen Tracking-API, ihre Definitionen, Wertetypen und Beispiele aufgeführt.

Eigenschaft	Definition	Werttyp	Beispiel
adID	- HLS — die Sequenznummer, die dem Anfang der Anzeige zugeordnet ist - DASH — die Perioden-ID der Anzeige	String	10
adBreakTrackingEvents	Ein Array, das VMAP-Tracking-Ereignisse aus der VAST-Antwort enthält. Weitere Informationen finden Sie in Abschnitt 2.3.3 der VMAP 1.0-Spezifikation.	String	[]

Eigenschaft	Definition	Werttyp	Beispiel
<code>adMarkerDuration</code>	Die anhand der Werbemarkierung im Manifest beobachtete Gültigkeitsdauer.	String	30
<code>adParameters</code>	Eine Reihe von Anzeigenparametern aus dem VAST VPAID, die an den MediaTailor Player weitergegeben wird.	String	
<code>adProgramDateTime</code>	- HLS — das Datum im Format ISO/IEC 8601:2004, das die erste Mediensequenz der Anzeige darstellt. - GEDANKENS TRICH -	String	
<code>ads</code>	Ein Array, das die Werbeobjekte enthält, aus denen sich der Avail zusammensetzt. Die Anzeigen werden in der Reihenfolge aufgeführt, in der sie im Manifest erscheinen.	Array	[]

Eigenschaft	Definition	Werttyp	Beispiel
adSystem	Der Name des Systems, das die Anzeige bereitstellt.  Important Stellen Sie sicher, dass Sie einen Wert angeben. Wenn Sie keinen Wert angeben, können Probleme auftreten.	String	myADS
adTitle	Der Titel der Anzeige.	String	ad1

Eigenschaft	Definition	Werttyp	Beispiel
adVerifications	Enthält die Ressource n und Metadaten, die für die Ausführung von Messcode von Drittanbietern erforderlich sind, um die kreative Wiedergabe zu überprüfen. Weitere Informationen zu dieser Eigenschaft finden Sie in Abschnitt 3.16 der VAST 4.2-Spezifikation. MediaTailor unterstützt adVerifications als VAST 3 Erweiterungsknoten.	Array	[]
altText	Der alternative Text für ein Bild einer Begleitanzeige. Dieser Text ermöglicht es Spielern mit beschreibender Audiounterstützung für sehbehinderte Personen, eine Beschreibung des Bildes vorzulesen.	String	video sequence advertising sneakers

Eigenschaft	Definition	Werttyp	Beispiel
<code>attributes</code>	Beinhaltet Schlüssel, die in der VAST-Spezifikation für Begleitwerbung definiert sind <code>adSlotIdpxratio</code> , <code>wierenderingMode</code> ,, usw.	Object	<code>{}</code>
<code>apiFramework</code>	Auf einstellen, VPAID um dem Player mitzuteilen, dass es sich bei dieser Anzeige um eine VPAID-Anzeige handelt.	String	VPAID
<code>availID</code>	- HLS — Die Sequenznummer, die mit dem Start der verfügbaren Anzeige verknüpft ist. - DASH — die Perioden-ID der Werbeanzeige, bei der es sich in der Regel um die Perioden-ID des Inhalts handelt, der durch eine Anzeige ersetzt werden soll.	String	34 - PT34S_1

Eigenschaft	Definition	Werttyp	Beispiel
<code>avails</code>	Ein Array, das Werbeblock-Objekte oder Avails enthält, die im aktiven Manifestfenster angezeigt werden. Die Avails werden in der Reihenfolge aufgeführt, in der sie im Manifest erscheinen.	Array	<code>[]</code>
<code>beaconUrls</code>	Die URL, an die der Ad Beacon MediaTailor gesendet wird.	String	
<code>bitrate</code>	Die Bitrate des Video-Assets. Diese Eigenschaft ist in der Regel nicht für ein ausführbares Asset enthalten.	String	<code>2048</code>

Eigenschaft	Definition	Werttyp	Beispiel
<code>companionAds</code>	Eine oder mehrere Spezifikationen für begleitende Anzeigeninhalte, von denen jede eine zu verwendende Ressourcendatei spezifiziert. Begleitanzeigen begleiten die Werbeanzeige und bieten Inhalte, wie z. B. einen Rahmen um die Anzeige oder ein Banner, der in der Nähe des Videos eingeblendet wird.	Array	<code>[]</code>
<code>companion ClickThrough</code>	Eine URL zur Seite des Werbetreibenden, die der Media Player öffnet, wenn der Zuschauer auf die Begleitanzeige klickt.	String	<code>https://aws.amazon.com/</code>
<code>companion ClickTracking</code>	Die Tracking-URL für die <code>companion ClickThrough</code> Immobilie.	String	<code>https://myads.com/beaconing/event=clicktracking</code>
<code>creativeId</code>	Der Id Attributwert des Creative Tags für die Anzeige.	String	<code>creative-1</code>

Eigenschaft	Definition	Werttyp	Beispiel
<code>creativeSequence</code>	Die Reihenfolge, in der eine Anzeige abgespielt werden soll, entsprechend dem Ad@id Wert in der VAST-Antwort.	String	1
<code>dashAvailabilityStartTime</code>	Bei Live/Dynamic DASH ist dies das MPD@availabilityStartTime Ursprungs-Manifest.	String	2022-10-05T19:38:39.263Z
<code>delivery</code>	Gibt an, ob ein progressive streaming OR-Protokoll verwendet wird.	String	progressive
<code>duration</code>	Länge im ISO-8601-Sekundenformat. Die Antwort umfasst die Dauer für die gesamte Anzeigenverfügbarkeit sowie für jede Anzeige und jedes Beacon, obwohl die Dauer der Beacons immer Null ist.	Anzahl	15.015

Eigenschaft	Definition	Werttyp	Beispiel
eventId	- HLS — die dem Beacon zugeordnete Sequenznummer. - DASH — der ptsTime Zeitpunkt des Starts der Anzeige.	String	23
eventType	Die Art des Beacons.	String	impression
extensions	Benutzerdefinierte Erweiterungen von VAST, die Werbeserver verwenden. Weitere Informationen zu Erweiterungen finden Sie in Abschnitt 3.18 der VAST 4.2-Spezifikation .	Array	[]
height	Die Höhe des Video-Assets in Pixeln.	String	360
hlsAnchorMediaSequenceNumber	Die Mediensequenznummer der ersten/ältesten Mediensequenz, die im HLS-Ursprungsmanifest zu finden ist.	String	77

Eigenschaft	Definition	Werttyp	Beispiel
<code>htmlResource</code>	Der CDATA-kodierte HTML-Code, der direkt in die HTML-Seite des Streaming-Anbieters eingefügt wird.	String	<code><![CDATA[<!doctype html><html <head><meta name=\"viewport \" content= \"width=1, initial-s cale=1.0, minimum-s cale=1.0, ...]]></code>
<code>iFrameResource</code>	Die URL zu einer HTML-Ressourcendatei, die der Streaming-Anbieter in einen Iframe lädt.	String	
<code>maintainAspectRatio</code>	Gibt an, ob das Seitenverhältnis des Videos bei der Skalierung beibehalten werden soll.	Boolesch	<code>true</code>
<code>mediaFilesList</code>	Gibt das Video und andere Elemente an, die der Player für die Anzeigenbereitstellung benötigt.	Array	<code>[]</code>
<code>mediaFileUri</code>	URI, die entweder auf ein ausführbares Asset oder ein Video-Asset verweist.	String	<code>https://myad.com/ad/ad134/vpaid.js</code>

Eigenschaft	Definition	Werttyp	Beispiel
mediaType	Der MIME-Typ des kreativen Elements oder des Begleitelements.	String	video/mp4
meta			
mezzanine	Die URL des MP4 Mezzanine-Assets, angegeben, falls die VPAID-Anzeige eines enthält.	String	https://gcdn.2mdn.net/videoplayback/id/itag/ck2/file/file.mp4
nextToken	Der Wert des Tokens, das auf die nächste Ergebnisseite verweist, sofern ein solcher Wert vorhanden ist.	String	UFQzOS44NzNTXzIwMjMtMDctMzFUMTY6NTA6MDYyMzUwNjI2ODQ1Wl8x
nonLinearAds		Array	[]
nonLinearAdsList		Array	[]
nonLinearAvails		Array	
scalable	Gibt an, ob das Video auf andere Dimensionen skaliert werden soll.	Boolesch	true
sequence	Der Sequenzwert, der für das Creative in der VAST-Antwort angegeben wurde.	String	1

Eigenschaft	Definition	Werttyp	Beispiel
<code>skipOffset</code>	Der Zeitwert, der angibt, wann der Player dem Benutzer Steuerelemente zum Überspringen zur Verfügung stellt.	String	<code>00:00:05</code>
<code>startTime</code>	Die Zeitposition im Format ISO 8601 Sekunden relativ zum Beginn der Wiedergabe-Sitzung. Die Antwort enthält Startzeiten für den gesamten Ad-Avail und für jede Werbung und jedes Beacon.	String	<code>PT9.943S</code>
<code>startTimeInSeconds</code>	Die Zeitposition im Sekundenformat relativ zum Beginn der Wiedergabesitzung. Die Antwort enthält Startzeiten für den gesamten Ad-Avail und für jede Werbung und jedes Beacon.	Anzahl	<code>9.943</code>
<code>staticResource</code>	Die URL zu einer statischen kreativen Datei, die für die Anzeigenkomponente verwendet wird.	String	<code>https://very-interactive-ads.com/campaign1/file.json?c=1019113602</code>

Eigenschaft	Definition	Werttyp	Beispiel
vastAdId	Der Id Attributwert des Ad Tags.	String	ad1
width	Die Breite des Video-Assets in Pixeln.	String	640

Zeitlicher Ablauf von Ad-Tracking-Aktivitäten

Bei der clientseitigen Berichterstattung muss der Spieler Tracking-Ereignisse (Beacons) mit einer gewissen Genauigkeit aussenden. Mithilfe des MediaTailor clientseitigen Tracking-Schemas können Sie sicherstellen, dass für alle Verfügbarkeits-, Anzeige-, Begleitungs-, Overlay- und Tracking-Ereignisse Informationen zum Zeitpunkt und zur Dauer vorliegen, und zwar in unterschiedlicher Form.

Verwenden Sie die folgenden MediaTailor Schlüssel/Wert-Paare für den Player, um Werbeereignisse wie das Verfolgen von Ereignissen genau mit der Wiedergabeposition in Einklang zu bringen:

- [startTime](#)
- [startTimeInSeconds](#)
- [adProgramDateTime](#)
- [adID/eventId](#)

HLS und DASH implementieren den Wert von und unterschiedlich: `startTime`
`startTimeInSeconds`

- HLS — Die `startTime` Werte beziehen sich auf den Beginn der Wiedergabe-Sitzung. Der Beginn der Wiedergabe-Sitzung ist als Zeit Null definiert. Bei der Anzeige `startTime` handelt es sich um die Summe der kumulierten Werte aller EXT-INF Segmentdauern, die bis zur Veröffentlichung geführt haben. Die Mediensequenznummer des Segments, auf das sich die Anzeige oder das Tracking-Ereignis bezieht, entspricht auch dem `adId` oder `eventId` in der clientseitigen Tracking-Antwort.
- DASH:
 - Live/dynamische Manifeste — Die `startTime` Werte beziehen sich auf die Werte des DASH-Manifests `MPD@availabilityStartTime`. Der `MPD@availabilityStartTime` ist ein Timing-Anker für alle MediaTailor Sessions, die den Stream nutzen.

- VOD/statische Manifeste — Die `startTime` Werte beziehen sich auf den Beginn der Wiedergabe-Sitzung. Der Beginn der Wiedergabesitzung ist als Zeitpunkt Null definiert. Jede Anzeige im Avail ist in einem eigenen `Period` Element enthalten. Das `Period` Element hat ein `@start` Attribut mit einem Wert, der den `startTime` Werten in der clientseitigen Tracking-Payload entspricht. Das entspricht `PeriodId` auch dem `adId` oder `eventId` in der clientseitigen Tracking-Antwort.

Example HLS:

Im folgenden Beispiel wurde die MediaTailor Sitzung gestartet, und das folgende Manifest ist das erste, das dem Client zugestellt wird:

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:6
#EXT-X-MEDIA-SEQUENCE:4603263
#EXT-X-DISCONTINUITY-SEQUENCE:0
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:23.295678Z
#EXTINF:4.010667,
https://123.cloudfront.net/out/v1/index_1_34.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:27.306345Z
#EXTINF:4.010667,
https://123.cloudfront.net/out/v1/index_1_35.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:31.317012Z
#EXTINF:4.010667,
https://123.cloudfront.net/out/v1/index_1_36.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:35.327679Z
#EXTINF:4.010667,
https://123.cloudfront.net/out/v1/index_1_37.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:39.338346Z
#EXTINF:2.538667,
https://123.cloudfront.net/out/v1/index_1_38.ts
#EXT-X-DISCONTINUITY
#EXT-X-KEY:METHOD=NONE
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:41.453Z
#EXTINF:2.0,
https://123.cloudfront.net/tm/asset_1080_4_8_00001.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:43.453Z
#EXTINF:2.0,
https://123.cloudfront.net/tm/asset_1080_4_8_00002.ts
#EXT-X-PROGRAM-DATE-TIME:2023-05-03T21:24:45.453Z
```



```
#EXTINF:2.0,
https://123.cloudfront.net/tm/asset_1080_4_8_00003.ts
```

In der JSON-Nutzlast für die clientseitige Nachverfolgung gelten die folgenden Werte:

- `startTime`: "PT18.581355S"
- `startTimeInSeconds`: 18.581
- `availProgramDateTime`: "2023-05-03T21:24:41.453Z"
- `adId`: 4603269

Example DASH:

Im folgenden Beispiel erhält die MediaTailor Sitzung eine Midroll im Manifest. Beachten Sie, dass der `@start` Attributwert der zweiten Periode, bei der es sich um den Anzeigenzeitraum handelt, einen Wert hat, der `MPD@availabilityStartTime` relativ zum Wert ist. Dieser Wert wird für alle Sitzungen in die clientseitigen `startTime` Tracking-Antwortfelder MediaTailor geschrieben.

```
<?xml version="1.0" encoding="UTF-8"?>
<MPD availabilityStartTime="2022-10-05T19:38:39.263Z" minBufferTime="PT10S"
  minimumUpdatePeriod="PT2S" profiles="urn:mpeg:dash:profile:isoff-live:2011"
  publishTime="2023-05-03T22:06:48.411Z" suggestedPresentationDelay="PT10S"
  timeShiftBufferDepth="PT1M30S" type="dynamic" xmlns="urn:mpeg:dash:schema:mpd:2011"
  xmlns:scte35="urn:scte:scte35:2013:xml" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
  instance" xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd">
  <BaseUrl>https://123.channel-assembly.mediataylor.us-west-2.amazonaws.com/v1/
  channel/my-channel/</BaseUrl>
  <Location>https://123.cloudfront.net/v1/
  dash/94063eadf7d8c56e9e2edd84fdf897826a70d0df/MediaTailor-Live-HLS-DASH/channel/
  channel1/dash.mpd?aws.sessionId=794a15e0-2a7f-4941-a537-9d71627984e5</Location>
  <Period id="1683151479166_1" start="PT5042H25M59.903S"
  xmlns="urn:mpeg:dash:schema:mpd:2011">
    <BaseUrl>https://123.cloudfront.net/out/v1/f1a946be8efa45b0931ea35c9055fb74/
    ddb73bf548a44551a0059c346226445a/eea5485198bf497284559efb8172425e/</BaseUrl>
    <AdaptationSet ...>
      ...
    </AdaptationSet>
  </Period>
  <Period id="1683151599194_1_1" start="PT5042H27M59.931S">
    <BaseUrl>https://123.cloudfront.net/
    tm/94063eadf7d8c56e9e2edd84fdf897826a70d0df/fpc5omz5wzd2rdepgieibp23ybyqyrme/</BaseUrl>
```

```
<AdaptationSet ...>
  ...
</AdaptationSet>
</Period>
</MPD>
```

In der JSON-Nutzlast für clientseitiges Tracking gelten die folgenden Werte:

- `startTime`: "PT5042H27M59.931S"
- `startTimeInSeconds`: 18152879.931
- `availProgramDateTime`: *null*
- `adId`: 1683151599194_1_1

Player-Steuerung und Funktionen für kundenseitiges Ad-Tracking

MediaTailor Die clientseitigen Tracking-Metadaten unterstützen verschiedene Player-Steuerelemente und Funktionen. In der folgenden Liste werden beliebte Player-Steuerelemente beschrieben.

Themen

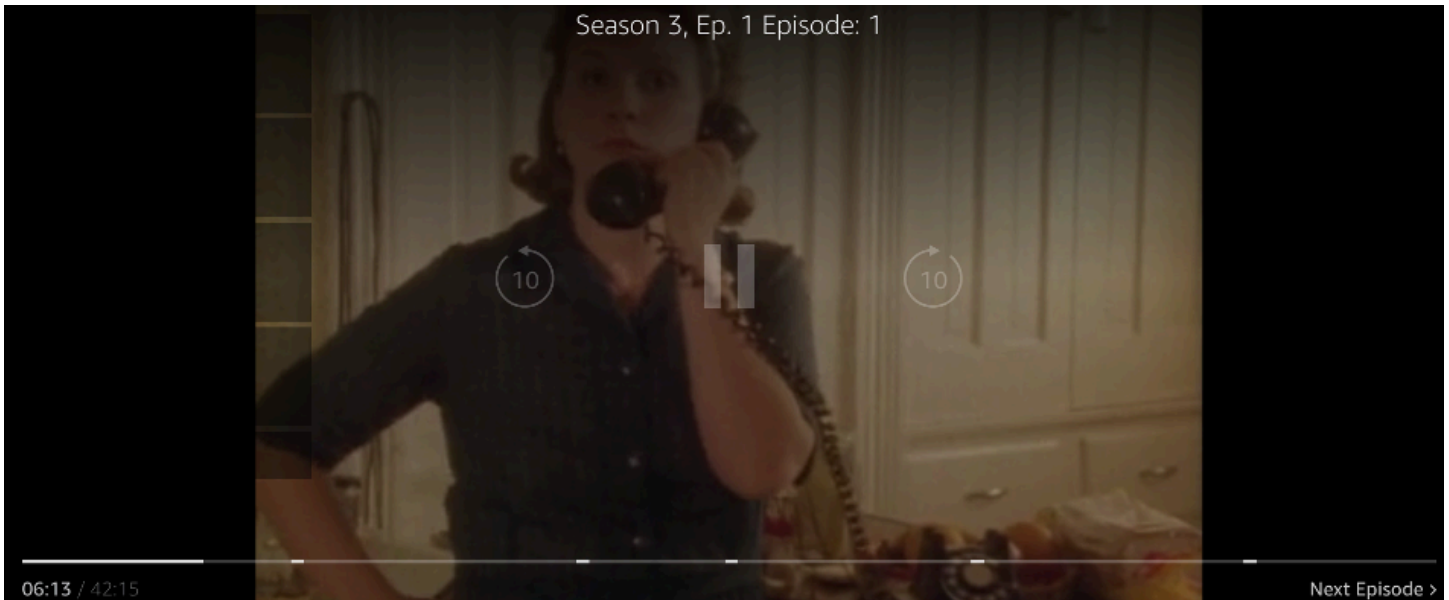
- [Schrubben](#)
- [Countdown-Timer hinzufügen](#)
- [Überspringbare Anzeigen](#)
- [Und Click-through](#)
- [Begleitende Anzeigen](#)
- [Interaktive Anzeigen \(SIMID\)](#)
- [Interaktive Werbung \(VPAID\)](#)
- [Symbole für Google Warum diese Anzeige \(WTA\)](#)

Schrubben

Um das Wiedergabeerlebnis zu verbessern, kann der Player Anzeigenpositionen in der Wiedergabe-Timeline anzeigen. MediaTailor stellt diese Anzeigenpositionen in Form von `adStartTimeInSeconds` Werten in der clientseitigen Tracking-Antwort zur Verfügung.

Note

Einige Streaming-Anbieter verhindern, dass Sie über eine Anzeigenposition hinausgehen.



Die folgende JSON-Antwort auf die clientseitige Tracking-Nutzlast zeigt die Startzeit der Verfügbarkeit (Werbeunterbrechung) innerhalb des JSON-Stammobjekts des Avails-Arrays. Der Player verwendet diese Daten, um den Ort der Werbeunterbrechung in der Player-Zeitleiste anzuzeigen, und zwar nach 28 Sekunden.

```
{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [...],
      "availId": "7",
      "availProgramDateTime": null,
      "duration": "PT30S",
      "durationInSeconds": 30,
      "meta": null,
      "nonLinearAdsList": [],
      "startTime": "PT28S",
      "startTimeInSeconds": 28
    }
  ],
}
```

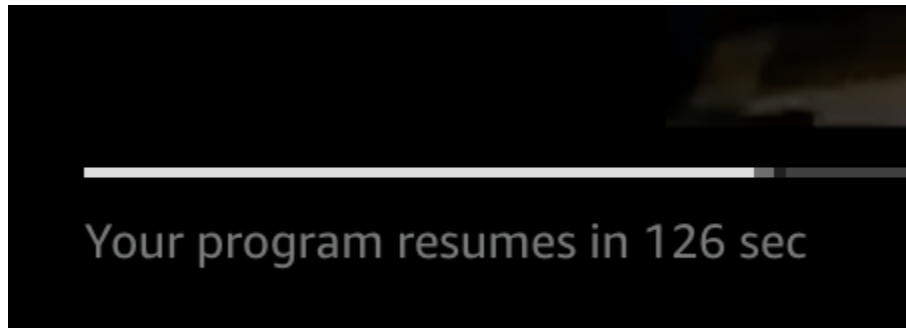
```

    "dashAvailabilityStartTime": null,
    "hlsAnchorMediaSequenceNumber": null,
    "nextToken": "UFQxMk0zNC44NjhTXzIwMjMtMDctMjFUMjA6MjM6MDcuNzc1NzE2MzAyWl8x",
    "nonLinearAvails": []
  }

```

Countdown-Timer hinzufügen

Mithilfe eines Countdown-Timers für Werbeanzeigen können MediaTailor Sie dafür sorgen, dass Ihr Publikum während der Werbepause bei der Stange bleibt. Anhand des Timers kann die Zielgruppe nachvollziehen, wann die Werbepause endet und ihr Programm wieder aufgenommen wird.



Die Elemente in den clientseitigen Tracking-Metadaten, die für den Countdown-Timer der Anzeige eine Rolle spielen, sind `startTime`, und `startTimeInSeconds` `duration` `durationInSeconds`. Der Player verwendet diese Metadaten zusammen mit der verstrichenen Sitzungszeit, die er separat verfolgt, um zu bestimmen, wann der Timer angezeigt werden soll und ab welchem Wert er herunterzählen soll.

Die folgende JSON-Antwort auf die clientseitige Tracking-Payload zeigt die Informationen, die für die Anzeige eines Countdown-Timers für Anzeigen erforderlich sind.

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [...],
      "availId": "7",
      "availProgramDateTime": null,
      "duration": "PT30S",
      "durationInSeconds": 30,
      "meta": null,
      "nonLinearAdsList": [],
      "startTime": "PT28S",

```

```

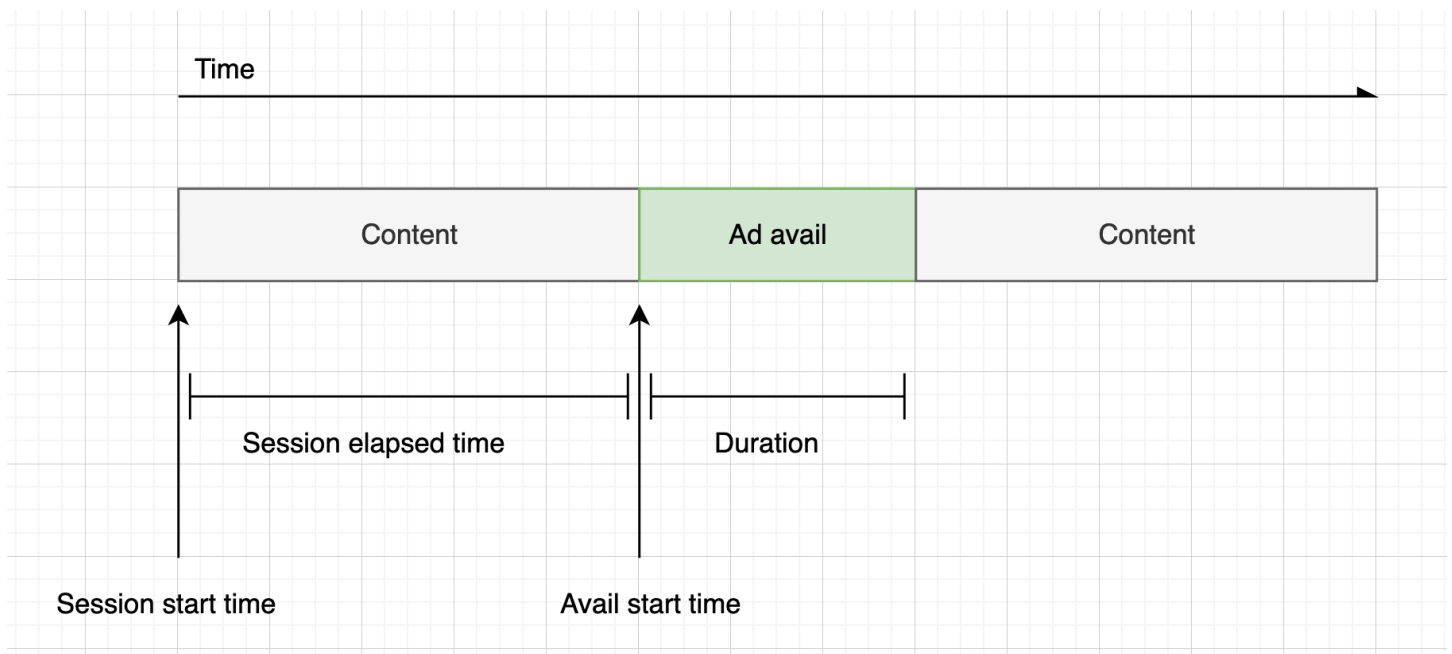
    "startTimeInSeconds": 28
  }
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQxMk0zNC44NjhTXzIwMjMtMDctMjFUMjA6MjM6MDcuNzc1NzE2MzAyWl8x",
"nonLinearAvails": []
}

```

Wenn die verstrichene Zeit der Sitzung die Startzeit des Avails erreicht, zeigt der Player einen Countdown-Timer mit einem Wert an, der der Dauer der Verfügbarkeit entspricht. Der Wert des Countdown-Timers nimmt ab, wenn die verstrichene Zeit die Startzeit des Avails überschreitet.

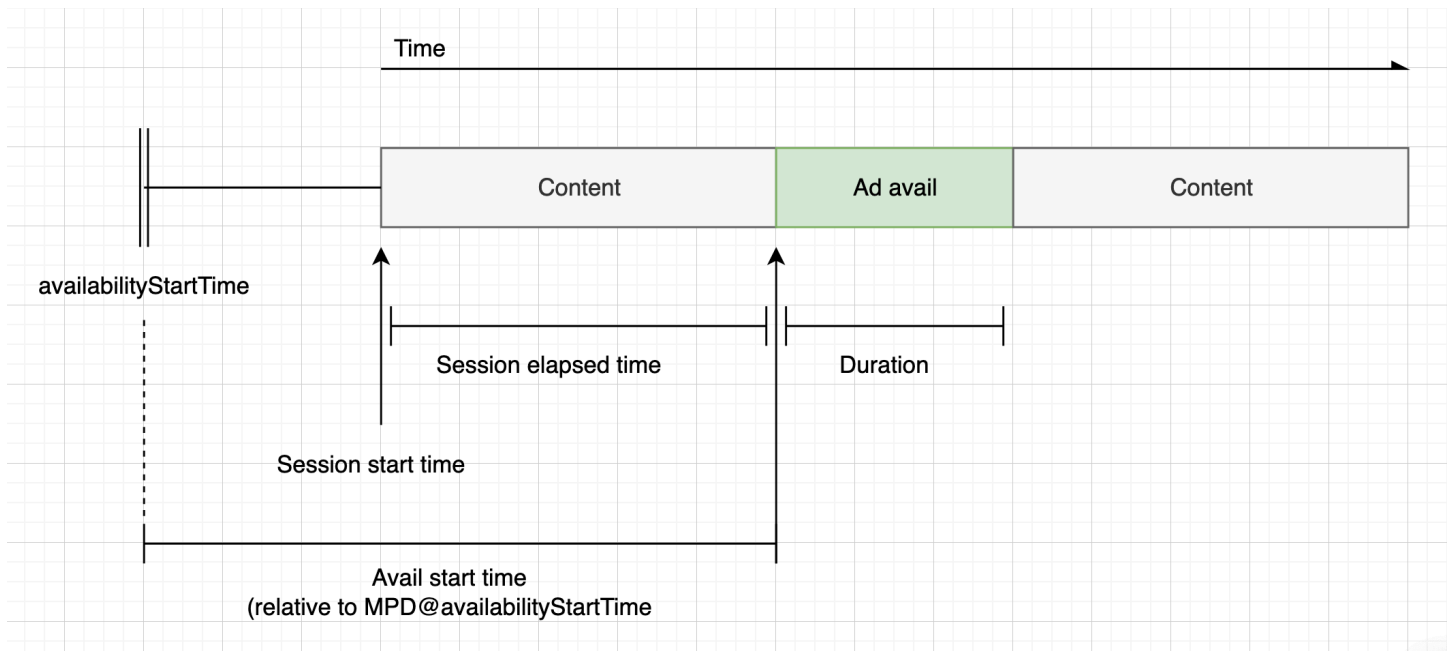
Example Formel: Countdown-Timer für HLS (Live und VOD) und DASH (VOD)

- $\text{session\_start\_time} = \text{die Summe aller EXT-INF Dauerwerte} - \text{der Dauerwert der drei neuesten Mediensequenzen EXT-INF}$
- $\text{Timer-Wert} = \text{duration} - (\text{session\_elapsed\_time} - \text{startTime})$



Example Formel: Countdown-Timer für DASH (live)

- $\text{session\_start\_time} = (\text{neuestes Segment startTime} + \text{duration}) / \text{timescale} - \text{MPD@suggestedPresentationDelay}$
- $\text{Timer-Wert} = \text{duration} - (\text{session\_elapsed\_time} - \text{startTime})$



Überspringbare Anzeigen

Überspringbare Anzeigen sind Werbespots, die es dem Zuschauer ermöglichen, einen Teil der Anzeige zu überspringen, um das Programm wieder anzusehen. In VAST identifiziert das `Linear@skipOffset` Attribut eine Anzeige, die übersprungen werden kann.

Die folgende VAST-Antwort zeigt, wie eine überspringbare Anzeige verwendet wird:

```
<?xml version="1.0" encoding="UTF-8"?>
<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad>
    <InLine>
      ...
    <Creatives>
      <Creative id="1" sequence="1">
        <Linear skipoffset="00:00:05">
          <Duration>00:00:15</Duration>
          <MediaFiles>
            <MediaFile id="EMT" delivery="progressive" width="640" height="360"
type="video/mp4" bitrate="143" scalable="true" maintainAspectRatio="true"><![CDATA[https://ads.com/file.mp4]]></MediaFile>
          </MediaFiles>
        </Linear>
      </Creative>
    </Creatives>
  </Ad>
</VAST>
```

```

    ...
  </InLine>
</Ad>
</VAST>

```

Die folgende JSON-Antwort für die clientseitige Tracking-Nutzlast zeigt die Anzeigenmetadaten innerhalb des Arrays. ads Das Array enthält den skipOffset Wert, der aus der MediaTailor VAST-Antwort abgerufen wurde.

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "1",
          "adParameters": "",
          "adProgramDateTime": "2023-07-31T16:11:40.693Z",
          "adSystem": "2.0",
          "adTitle": "AD-skiing-15",
          "adVerifications": [],
          "companionAds": [...],
          "creativeId": "1",
          "creativeSequence": "1",
          "duration": "PT15.015S",
          "durationInSeconds": 15.015,
          "extensions": [],
          "mediaFiles": {
            "mediaFilesList": [],
            "mezzanine": ""
          },
          "skipOffset": "00:00:05",
          "startTime": "PT9.943S",
          "startTimeInSeconds": 9.943,
          "trackingEvents": [
            {
              "beaconUrls": [
                "https://adserverbeaconing.com/v1/impression"
              ],
              "duration": "PT15.015S",
              "durationInSeconds": 15.015,
              "eventId": "2697726",

```

```

        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT9.943S",
        "startTimeInSeconds": 9.943
      }
    ],
    "vastAdId": ""
  }
],
"availId": "2697726",
"availProgramDateTime": "2023-07-31T16:11:40.693Z",
"duration": "PT15.015S",
"durationInSeconds": 15.015,
"meta": null,
"nonLinearAdsList": [],
"startTime": "PT9.943S",
"startTimeInSeconds": 9.943
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "",
"nonLinearAvails": []
}

```

Und Click-through

Mithilfe von Click-Through URIs können Werbetreibende messen, wie erfolgreich eine Anzeige die Aufmerksamkeit der Zuschauer auf sich zieht. Nachdem ein Zuschauer auf den aktiven Videoframe einer laufenden Anzeige geklickt hat, öffnet ein Webbrowser den URI für die Startseite oder Kampagnen-Landingpage des Werbetreibenden. Der Player-Entwickler bestimmt das Klickverhalten, z. B. das Überlagern einer Schaltfläche oder eines Labels auf dem Werbevideo mit einer Meldung, auf die Sie klicken müssen, um mehr zu erfahren. Player-Entwickler pausieren das Video der Anzeige häufig, nachdem Zuschauer auf den aktiven Videoframe geklickt haben.



Click here for deals on Amazon.com

MediaTailor kann jedes lineare Video-Klick-Through-Ereignis analysieren und verfügbar machen, das in der VAST-Antwort URLs zurückgegeben wurde. Die folgende VAST-Antwort zeigt ein Beispiel für einen Klick auf eine Anzeige.

```
<?xml version="1.0" encoding="UTF-8"?>
<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad>
    <Inline>
      ...
    <Creatives>
      <Creative id="1" sequence="1">
        <Linear>
          <Duration>00:00:15</Duration>
          <MediaFiles>
```

```

        <MediaFile id="EMT" delivery="progressive" width="1280" height="720"
        type="video/mp4" bitrate="143" scalable="true" maintainAspectRatio="true"><![
[CDATA[https://ads.com/file.mp4]]></MediaFile>
        </MediaFiles>
        <VideoClicks>
            <ClickThrough id="EMT"><![CDATA[https://aws.amazon.com]]></ClickThrough>
            <ClickTracking id="EMT"><![CDATA[https://myads.com/beaconing/
event=clicktracking]]></ClickTracking>
        </VideoClicks>
    </Linear>
</Creative>
</Creatives>
...
</InLine>
</Ad>
</VAST>

```

Die folgende JSON-Antwort für die clientseitige Tracking-Payload zeigt, wie Click-Through und Click-Tracking MediaTailor innerhalb des Arrays dargestellt werden. URLs `trackingEvents` Der `clickThrough` Ereignistyp steht für die Click-Through-Anzeige, und der Ereignistyp steht für die Click-Tracking-URL. `clickTracking`

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "1",
          "adParameters": "",
          "adProgramDateTime": "2023-07-31T16:53:40.577Z",
          "adSystem": "2.0",
          "adTitle": "1",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "00006",
          "creativeSequence": "1",
          "duration": "PT14.982S",
          "durationInSeconds": 14.982,
          "extensions": [],
          "mediaFiles": {
            "mediaFilesList": [],

```

```
    "mezzanine": ""
  },
  "skipOffset": null,
  "startTime": "PT39.339S",
  "startTimeInSeconds": 39.339,
  "trackingEvents": [
    {
      "beaconUrls": [
        "https://myads.com/beaconing/event=impression"
      ],
      "duration": "PT14.982S",
      "durationInSeconds": 14.982,
      "eventId": "2698188",
      "eventProgramDateTime": null,
      "eventType": "impression",
      "startTime": "PT39.339S",
      "startTimeInSeconds": 39.339
    },
    {
      "beaconUrls": [
        "https://aws.amazon.com"
      ],
      "duration": "PT14.982S",
      "durationInSeconds": 14.982,
      "eventId": "2698188",
      "eventProgramDateTime": null,
      "eventType": "clickThrough",
      "startTime": "PT39.339S",
      "startTimeInSeconds": 39.339
    },
    {
      "beaconUrls": [
        "https://myads.com/beaconing/event=clicktracking"
      ],
      "duration": "PT14.982S",
      "durationInSeconds": 14.982,
      "eventId": "2698795",
      "eventProgramDateTime": null,
      "eventType": "clickTracking",
      "startTime": "PT39.339S",
      "startTimeInSeconds": 39.339
    }
  ],
  "vastAdId": ""
```

```

    }
  ],
  "availId": "2698188",
  "availProgramDateTime": "2023-07-31T16:53:40.577Z",
  "duration": "PT14.982S",
  "durationInSeconds": 14.982,
  "meta": null,
  "nonLinearAdsList": [],
  "startTime": "PT39.339S",
  "startTimeInSeconds": 39.339
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQz0S4zMz1TXzIwMjMtMDctMzFUMTY6NTQ6MDQuODA1Mzk2NTI5Wl8x",
"nonLinearAvails": []
}

```

Begleitende Anzeigen

Eine Begleitanzeige erscheint zusammen mit einem linearen Werbemittel. Verwenden Sie Begleitanzeigen, um die Effektivität eines Werbespots zu erhöhen, indem Sie Produkt-, Logo- und Markeninformationen anzeigen. Die Display-Anzeige kann QR-Codes (Quick Response) und anklickbare Bereiche enthalten, um das Engagement der Zuschauer zu fördern.

MediaTailor unterstützt Begleitanzeigen in der VAST-Antwort. Es kann Metadaten von `StaticResource` oder `FrameResource`, bzw. `HTMLResource` Knoten übertragen.

Die folgende VAST-Antwort zeigt ein Beispiel für Position und Format der linearen Anzeige und der Begleitanzeige.

```

<?xml version="1.0" encoding="UTF-8"?>
<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad>
    <Inline>
      ...
    <Creatives>
      <Creative id="1" sequence="1">
        <Linear>
          <Duration>00:00:10</Duration>
          <MediaFiles>

```

```

        <MediaFile id="EMT" delivery="progressive" width="640" height="360"
        type="video/mp4" bitrate="143" scalable="true" maintainAspectRatio="true"><![
[CDATA[https://ads.com/file.mp4]]></MediaFile>
    </MediaFiles>
</Linear>
</Creative>
<Creative id="2" sequence="1">
    <CompanionAds>
        <Companion id="2" width="300" height="250">
            <StaticResource creativeType="image/png"><![CDATA[https://emt.com/
companion/9973499273]]></StaticResource>
            <TrackingEvents>
                <Tracking event="creativeView"><![CDATA[https://beacon.com/1]]></
Tracking>
            </TrackingEvents>
            <CompanionClickThrough><![CDATA[https://beacon.com/2]]></
CompanionClickThrough>
        </Companion>
        <Companion id="3" width="728" height="90">
            <StaticResource creativeType="image/png"><![CDATA[https://emt.com/
companion/1238901823]]></StaticResource>
            <TrackingEvents>
                <Tracking event="creativeView"><![CDATA[https://beacon.com/3]]></
Tracking>
            </TrackingEvents>
            <CompanionClickThrough><![CDATA[https://beacon.com/4]]></
CompanionClickThrough>
        </Companion>
    </CompanionAds>
</Creative>
</Creatives>
    ...
</InLine>
</Ad>
</VAST>

```

Die Daten werden in der clientseitigen Tracking-Antwort in der `/avail/x/ads/y/companionAds` Liste angezeigt. Jedes lineare Werbemittel kann bis zu 6 Begleitanzeigen enthalten. Wie im Beispiel unten gezeigt, werden die Begleitanzeigen in einer Liste angezeigt

 Note

Als bewährte Methode sollten Anwendungsentwickler eine Logik implementieren, mit der die Begleitanzeige am Ende der Werbeaktion explizit entfernt oder entfernt wird.

```
{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "0",
          "adParameters": "",
          "adProgramDateTime": null,
          "adSystem": "EMT",
          "adTitle": "sample",
          "adVerifications": [],
          "companionAds": [
            {
              "adParameters": null,
              "altText": null,
              "attributes": {
                "adSlotId": null,
                "apiFramework": null,
                "assetHeight": null,
                "assetWidth": null,
                "expandedHeight": null,
                "expandedWidth": null,
                "height": "250",
                "id": "2",
                "pxratio": null,
                "renderingMode": null,
                "width": "300"
              },
              "companionClickThrough": "https://beacon.com/2",
              "companionClickTracking": null,
              "htmlResource": null,
              "iFrameResource": null,
              "sequence": "1",
              "staticResource": "https://emt.com/companion/9973499273",
            }
          ],
          "sequence": "1",
          "staticResource": "https://emt.com/companion/9973499273",
        }
      ],
      "sequence": "1",
      "staticResource": "https://emt.com/companion/9973499273",
    }
  ],
  "sequence": "1",
  "staticResource": "https://emt.com/companion/9973499273",
}
```

```
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://beacon.com/1"
        ],
        "eventType": "creativeView"
      }
    ],
  },
  {
    "adParameters": null,
    "altText": null,
    "attributes": {
      "adSlotId": null,
      "apiFramework": null,
      "assetHeight": null,
      "assetWidth": null,
      "expandedHeight": null,
      "expandedWidth": null,
      "height": "90",
      "id": "3",
      "pxratio": null,
      "renderingMode": null,
      "width": "728"
    },
    "companionClickThrough": "https://beacon.com/4",
    "companionClickTracking": null,
    "htmlResource": null,
    "iFrameResource": null,
    "sequence": "1",
    "staticResource": "https://emt.com/companion/1238901823",
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://beacon.com/3"
        ],
        "eventType": "creativeView"
      }
    ]
  }
],
"creativeId": "1",
"creativeSequence": "1",
"duration": "PT10S",
```

```

    "durationInSeconds": 10,
    "extensions": [],
    "mediaFiles": {
      "mediaFilesList": [],
      "mezzanine": ""
    },
    "skipOffset": null,
    "startTime": "PT0S",
    "startTimeInSeconds": 0,
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://beacon.com/impression/1"
        ],
        "duration": "PT10S",
        "durationInSeconds": 10,
        "eventId": "0",
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT0S",
        "startTimeInSeconds": 0
      }
    ],
    "vastAdId": ""
  }
],
"availId": "0",
"availProgramDateTime": null,
"duration": "PT10S",
"durationInSeconds": 10,
"meta": null,
"nonLinearAdsList": [],
"startTime": "PT0S",
"startTimeInSeconds": 0
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQxMFNfMjAyMy0wNy0wNlQyMT0xMDowOC42NzQ4NDA1NjJaXzE%3D",
"nonLinearAvails": []
}

```


Interaktive Anzeigen (SIMID)

SecureInteractive Media Interface Definition (SIMID) ist ein Standard für interaktive Werbung, der im VAST 4.x-Standard des Interactive Advertising Bureau (IAB) eingeführt wurde. SIMID entkoppelt das Laden interaktiver Elemente vom primären linearen Werbematerial auf dem Player und bezieht sich in der VAST-Antwort auf beide. MediaTailor fügt das primäre Creative hinzu, um das Wiedergabeerlebnis aufrechtzuerhalten, und platziert Metadaten für die interaktiven Komponenten in der clientseitigen Tracking-Antwort.

In der folgenden VAST-4-Beispielantwort befindet sich die SIMID-Nutzlast innerhalb des Knotens `InteractiveCreativeFile`

```
<?xml version="1.0"?>
<VAST xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad id="1234567">
    <Inline>
      <AdSystem>SampleAdSystem</AdSystem>
      <AdTitle>Linear SIMID Example</AdTitle>
      <Description>SIMID example</Description>
      <Error>https://www.beacons.com/error</Error>
      <Impression>https://www.beacons.com/impression</Impression>
      <Creatives>
        <Creative sequence="1">
          <Linear>
            <Duration>00:00:15</Duration>
            <TrackingEvents>
              ...
            </TrackingEvents>
            <VideoClicks>
              <ClickThrough id="123">https://aws.amazon.com</ClickThrough>
              <ClickTracking id="123">https://www.beacons.com/click</ClickTracking>
            </VideoClicks>
            <MediaFiles>
              <MediaFile delivery="progressive" type="video/mp4">
https://interactive-ads.com/interactive-media-ad-sample/media/file.mp4
              </MediaFile>
              <InteractiveCreativeFile type="text/html" apiFramework="SIMID"
                variableDuration="true">
https://interactive-ads.com/interactive-media-ad-sample/sample\_simid.html
              </InteractiveCreativeFile>
```

```

        </MediaFiles>
    </Linear>
</Creative>
</Creatives>
</InLine>
</Ad>
</VAST>

```

In der folgenden VAST-3-Antwort befindet sich die SIMID-Nutzlast innerhalb des Knotens. Extensions

```

<?xml version="1.0"?>
<VAST xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad id="1234567">
    <InLine>
      <AdSystem>SampleAdSystem</AdSystem>
      <AdTitle>Linear SIMID Example</AdTitle>
      <Description>SIMID example</Description>
      <Impression>https://www.beacons.com/impression</Impression>
      <Creatives>
        <Creative id="1" sequence="1">
          <Linear>
            <Duration>00:00:15</Duration>
            <TrackingEvents>
              ...
            </TrackingEvents>
            <VideoClicks>
              <ClickThrough id="123">https://aws.amazon.com</ClickThrough>
              <ClickTracking id="123">https://myads.com/beaconing/event=clicktracking</
ClickTracking>
            </VideoClicks>
            <MediaFiles>
              <MediaFile delivery="progressive" type="video/mp4">
https://interactive-ads.com/interactive-media-ad-
sample/media/file.mp4
              </MediaFile>
            </MediaFiles>
          </Linear>
        </Creative>
      </Creatives>
      <Extensions>
        <Extension type="InteractiveCreativeFile">

```

```

        <InteractiveCreativeFile type="text/html" apiFramework="SIMID"
variableDuration="true">
            https://interactive-ads.com/interactive-media-ad-sample/sample_simid.html
        </InteractiveCreativeFile>
    </Extension>
</Extensions>
</InLine>
</Ad>
</VAST>

```

In der folgenden clientseitigen Tracking-Antwort werden die SIMID-Daten in der Liste angezeigt. /
 avails/x/ads/y/extensions

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "1",
          "adParameters": "",
          "adProgramDateTime": "2023-07-31T16:53:40.577Z",
          "adSystem": "2.0",
          "adTitle": "Linear SIMID Example",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "1",
          "creativeSequence": "1",
          "duration": "PT14.982S",
          "durationInSeconds": 14.982,
          "extensions": [
            {
              "content": "<InteractiveCreativeFile type=\"text/html\" apiFramework=
\\\"SIMID\\\" variableDuration=\\\"true\\\">\nhttps://interactive-ads.com/interactive-media-ad-
sample/sample_simid.html</InteractiveCreativeFile>",
              "type": "InteractiveCreativeFile"
            }
          ],
          "mediaFiles": {
            "mediaFilesList": [],
            "mezzanine": ""
          }
        }
      ]
    }
  ]
}

```

```
"skipOffset": null,
"startTime": "PT39.339S",
"startTimeInSeconds": 39.339,
"trackingEvents": [
  {
    "beaconUrls": [
      "https://myads.com/beaconing/event=impression"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698188",
    "eventProgramDateTime": null,
    "eventType": "impression",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  },
  {
    "beaconUrls": [
      "https://aws.amazon.com"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698188",
    "eventProgramDateTime": null,
    "eventType": "clickThrough",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  },
  {
    "beaconUrls": [
      "https://myads.com/beaconing/event=clicktracking"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698795",
    "eventProgramDateTime": null,
    "eventType": "clickTracking",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  }
],
"vastAdId": ""
}
```

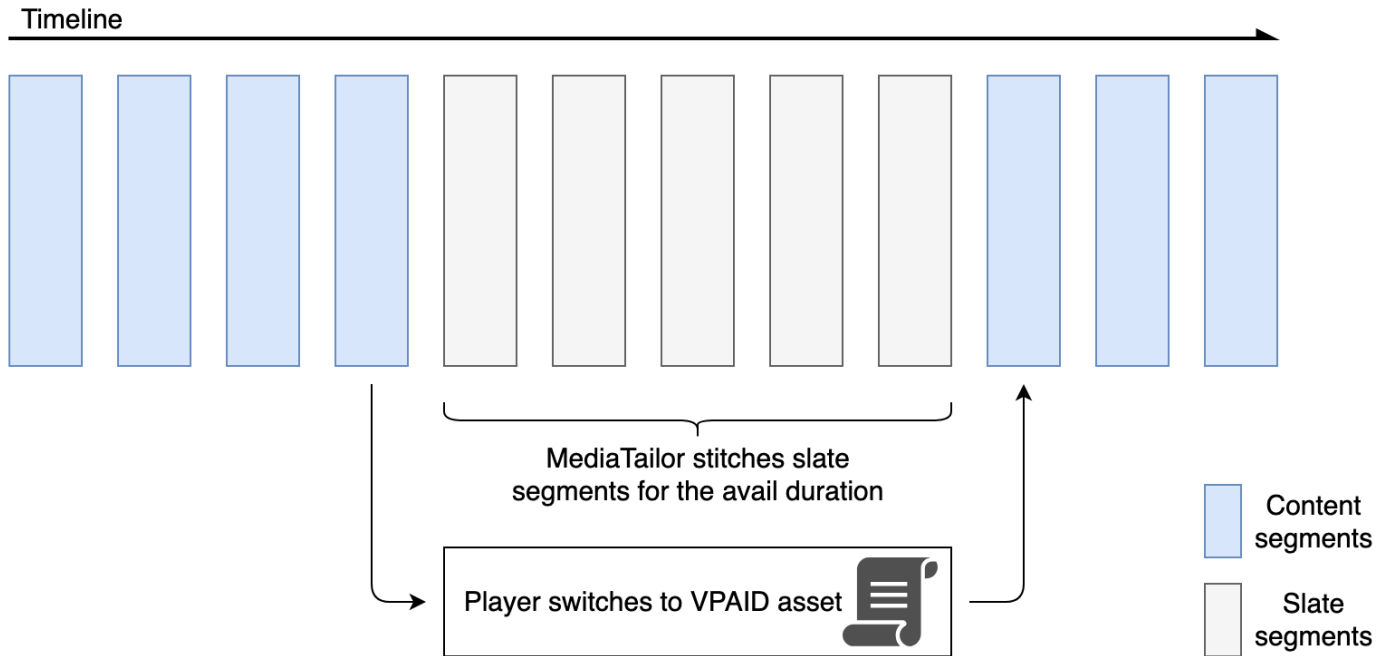
```
    "availId": "2698188",
    "availProgramDateTime": "2023-07-31T16:53:40.577Z",
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "meta": null,
    "nonLinearAdsList": [],
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  }
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQz0S4zMz1TXzIwMjMtMDctMzFUMTY6NTQ6MDQu0DA1Mzk2NTI5Wl8x",
"nonLinearAvails": []
}
```

Interaktive Werbung (VPAID)

Die Video Player Ad Interface Definition (VPAID) spezifiziert das Protokoll zwischen der Anzeige und dem Videoplayer, das Anzeigeninteraktivität und andere Funktionen ermöglicht. MediaTailor unterstützt bei Live-Streams das VPAID-Format, indem Slate-Segmente für die Dauer des Angebots zusammengefügt und Metadaten für die VPAID-Kreativen in die clientseitige Tracking-Antwort eingefügt werden, die der Videoplayer verarbeitet. Der Player lädt die VPAID-Dateien herunter, spielt das lineare Creative ab und führt die Skripte des Kunden aus. Der Spieler sollte niemals die Slate-Segmente spielen.

Note

VPAID ist ab VAST 4.1 veraltet.



Das folgende Beispiel zeigt den VPAID-Inhalt in der VAST-Antwort.

```
<?xml version="1.0"?>
<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad id="1234567">
    <Inline>
      <AdSystem>GDFP</AdSystem>
      <AdTitle>VPAID</AdTitle>
      <Description>Vpaid Linear Video Ad</Description>
      <Error>http://www.example.com/error</Error>
      <Impression>http://www.example.com/impression</Impression>
      <Creatives>
        <Creative sequence="1">
          <Linear>
            <Duration>00:00:00</Duration>
            <TrackingEvents>
              <Tracking event="start">http://www.example.com/start</Tracking>
              <Tracking event="firstQuartile">http://www.example.com/firstQuartile</
Tracking>
              <Tracking event="midpoint">http://www.example.com/midpoint</Tracking>
              <Tracking event="thirdQuartile">http://www.example.com/thirdQuartile</
Tracking>
```

```

    <Tracking event="complete">http://www.example.com/complete</Tracking>
    <Tracking event="mute">http://www.example.com/mute</Tracking>
    <Tracking event="unmute">http://www.example.com/unmute</Tracking>
    <Tracking event="rewind">http://www.example.com/rewind</Tracking>
    <Tracking event="pause">http://www.example.com/pause</Tracking>
    <Tracking event="resume">http://www.example.com/resume</Tracking>
    <Tracking event="fullscreen">http://www.example.com/fullscreen</Tracking>
    <Tracking event="creativeView">http://www.example.com/creativeView</
Tracking>
    <Tracking event="acceptInvitation">http://www.example.com/
acceptInvitation</Tracking>
  </TrackingEvents>
  <AdParameters><![CDATA[ {"videos":[ {"url":"https://my-ads.com/interactive-
media-ads/media/media_linear_VPAID.mp4","mimetype":"video/mp4"}]] ]]></AdParameters>
  <VideoClicks>
    <ClickThrough id="123">http://google.com</ClickThrough>
    <ClickTracking id="123">http://www.example.com/click</ClickTracking>
  </VideoClicks>
  <MediaFiles>
    <MediaFile delivery="progressive" apiFramework="VPAID" type="application/
javascript" width="640" height="480"> https://googleads.github.io/googleads-ima-html5/
vpaid/linear/VpaidVideoAd.js </MediaFile>
  </MediaFiles>
</Linear>
</Creative>
</Creatives>
</InLine>
</Ad>
</VAST>

```

Das folgende Beispiel zeigt die Tracking-Informationen.

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "1",
          "adParameters": "",
          "adProgramDateTime": "2023-07-31T16:53:40.577Z",
          "adSystem": "2.0",

```

```
"adTitle": "1",
"adVerifications": [],
"companionAds": [],
"creativeId": "00006",
"creativeSequence": "1",
"duration": "PT14.982S",
"durationInSeconds": 14.982,
"extensions": [],
"mediaFiles": {
  "mediaFilesList": [],
  "mezzanine": ""
},
"skipOffset": null,
"startTime": "PT39.339S",
"startTimeInSeconds": 39.339,
"trackingEvents": [
  {
    "beaconUrls": [
      "https://myads.com/beaconing/event=impression"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698188",
    "eventProgramDateTime": null,
    "eventType": "impression",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  },
  {
    "beaconUrls": [
      "https://aws.amazon.com"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698188",
    "eventProgramDateTime": null,
    "eventType": "clickThrough",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  },
  {
    "beaconUrls": [
      "https://myads.com/beaconing/event=clicktracking"
    ],
```



```

        "duration": "PT14.982S",
        "durationInSeconds": 14.982,
        "eventId": "2698795",
        "eventProgramDateTime": null,
        "eventType": "clickTracking",
        "startTime": "PT39.339S",
        "startTimeInSeconds": 39.339
    }
],
    "vastAdId": ""
}
],
    "availId": "2698188",
    "availProgramDateTime": "2023-07-31T16:53:40.577Z",
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "meta": null,
    "nonLinearAdsList": [],
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
}
],
    "dashAvailabilityStartTime": null,
    "hlsAnchorMediaSequenceNumber": null,
    "nextToken": "UFQz0S4zMz1TXzIwMjMtMDctMzFUMTY6NTQ6MDQu0DA1Mzk2NTI5Wl8x",
    "nonLinearAvails": []
}
{
    "avails": [
        {
            "adBreakTrackingEvents": [],
            "adMarkerDuration": null,
            "ads": [
                {
                    "adId": "2922274",
                    "adParameters": "",
                    "adProgramDateTime": "2023-08-14T19:49:53.998Z",
                    "adSystem": "Innovid Ads",
                    "adTitle": "VPAID",
                    "adVerifications": [],
                    "companionAds": [],
                    "creativeId": "",
                    "creativeSequence": "",
                    "duration": "PT16.016S",
                    "durationInSeconds": 16.016,

```

```

    "extensions": [],
    "mediaFiles": {
      "mediaFilesList": [
        {
          "apiFramework": "VPAID",
          "bitrate": 0,
          "codec": null,
          "delivery": "progressive",
          "height": 9,
          "id": "",
          "maintainAspectRatio": false,
          "maxBitrate": 0,
          "mediaFileUri": "http://my-ads.com/mobileapps/js/vpaid/1h41kg?
cb=178344c0-8e67-281a-58ca-962e4987cd60&deviceid=&ivc=",
          "mediaType": "application/javascript",
          "minBitrate": 0,
          "scalable": false,
          "width": 16
        }
      ],
      "mezzanine": "http://my-ads.com/mobileapps/js/vpaid/1h41kg?
cb=178344c0-8e67-281a-58ca-962e4987cd60&deviceid=&ivc="
    },
    "skipOffset": null,
    "startTime": "PT8M42.289S",
    "startTimeInSeconds": 522.289,
    "trackingEvents": [
      {
        "beaconUrls": [
          "about:blank"
        ],
        "duration": "PT16.016S",
        "durationInSeconds": 16.016,
        "eventId": "2922274",
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT8M42.289S",
        "startTimeInSeconds": 522.289
      }
    ],
    "vastAdId": "1h41kg"
  }
],
  "availId": "2922274",

```

```

    "availProgramDateTime": "2023-08-14T19:49:53.998Z",
    "duration": "PT16.016S",
    "durationInSeconds": 16.016,
    "meta": null,
    "nonLinearAdsList": [],
    "startTime": "PT8M42.289S",
    "startTimeInSeconds": 522.289
  }
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQ4TTQyLjI4OVNfMjAyMy0wOC0xNFQxOT01MDo0MS4zOTc5MjAzODVaXzE%3D",
"nonLinearAvails": []
}

```

Symbole für Google Warum diese Anzeige (WTA)

AdChoices ist ein Industriestandard, der Zuschauern Informationen über die Anzeigen bietet, die sie sehen, einschließlich der Art und Weise, wie diese Anzeigen auf sie ausgerichtet wurden.



Die MediaTailor clientseitige Tracking-API unterstützt Symbolmetadaten, die im VAST-Erweiterungsknoten der VAST-Antwort gespeichert sind. Weitere Informationen zu WTA in der VAST-Antwort finden Sie in [dieser VAST-XML-Beispielantwort](#).

Note

MediaTailor unterstützt derzeit nur VAST Version 3.

```

<VAST>
  <Ad>
    <Inline>
      ...
    <Extensions>
      <Extension type="IconClickFallbackImages">

```

```

    <IconClickFallbackImages program="GoogleWhyThisAd">
      <IconClickFallbackImage width="400" height="150">
        <AltText>Alt icon fallback</AltText>
        <StaticResource creativeType="image/png"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png]]></StaticResource>
      </IconClickFallbackImage>
    </IconClickFallbackImages>
    <IconClickFallbackImages program="AdChoices">
      <IconClickFallbackImage width="400" height="150">
        <AltText>Alt icon fallback</AltText>
        <StaticResource creativeType="image/png"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png?size=1x]]></
StaticResource>
      </IconClickFallbackImage>
      <IconClickFallbackImage width="800" height="300">
        <AltText>Alt icon fallback</AltText>
        <StaticResource creativeType="image/png"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png?size=2x]]></
StaticResource>
      </IconClickFallbackImage>
    </IconClickFallbackImages>
  </Extension>
</Extensions>
</InLine>
</Ad>
</VAST>

```

Das folgende Beispiel zeigt die clientseitige Tracking-Antwort in der /avails/x/ads/y/extensions Liste.

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "0",
          "adParameters": "",
          "adProgramDateTime": null,
          "adSystem": "GDFP",
          "adTitle": "Google Why This Ad VAST 3 Sample",
          "adVerifications": [],

```

```

    "companionAds": [],
    "creativeId": "7891011",
    "creativeSequence": "1",
    "duration": "PT10S",
    "durationInSeconds": 10,
    "extensions": [
      {
        "content": "<IconClickFallbackImages program=\"GoogleWhyThisAd\">
          <IconClickFallbackImage height=\"150\" width=\"400\">
            <AltText>Alt icon fallback</AltText>
            <StaticResource creativeType=\"image/png\"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png]]>
          </StaticResource>
        </IconClickFallbackImage>
      </IconClickFallbackImages>
      <IconClickFallbackImages program=\"AdChoices\">
        <IconClickFallbackImage height=\"150\" width=\"400\">
          <AltText>Alt icon fallback</AltText>
          <StaticResource creativeType=\"image/png\"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png?size=1x]]>
        </StaticResource>
      </IconClickFallbackImage>
      <IconClickFallbackImage height=\"300\" width=\"800\">
        <AltText>Alt icon fallback</AltText>
        <StaticResource creativeType=\"image/png\"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png?size=2x]]>
        </StaticResource>
      </IconClickFallbackImage>
    </IconClickFallbackImages>",
        "type": "IconClickFallbackImages"
      }
    ],
    "mediaFiles": {
      "mediaFilesList": [],
      "mezzanine": ""
    },
    "skipOffset": "00:00:03",
    "startTime": "PT0S",
    "startTimeInSeconds": 0,
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://example.com/view"
        ]
      }
    ],

```

```

        "duration": "PT10S",
        "durationInSeconds": 10,
        "eventId": "0",
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT0S",
        "startTimeInSeconds": 0
      }
    ],
    "vastAdId": "123456"
  }
],
"availId": "0",
"availProgramDateTime": null,
"duration": "PT10S",
"durationInSeconds": 10,
"meta": null,
"nonLinearAdsList": [],
"startTime": "PT0S",
"startTimeInSeconds": 0
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQxMFNfMjAyMy0wNy0wNlQyMDo0MT0xNy45NDE4MDM0NDhaXzE%3D",
"nonLinearAvails": []
}

```

Clientseitiges Beaconsing

Mit dem clientseitigen `startTimeInSeconds` Tracking-Element können Sie MediaTailor das Timing von Beacons unterstützen.

Die folgende JSON-Antwort zeigt die wichtigsten Beacon-Typen: Impressionen, Start, Quartile und Abschluss.

Note

In den Richtlinien des Interactive Advertising Bureau (IAB) zur Messung von Videoimpressionen heißt es, dass für eine Impression der Anzeigeninhalt clientseitig geladen werden muss und mindestens die Startzeit für das Rendern im Player erreicht werden muss.

Weitere Informationen finden Sie unter [Digital Video Ad Serving Template \(VAST\)](#) auf der IAB-Website.

```
{
  "avails": [
    {
      "ads": [
        {
          "adId": "8104385",
          "duration": "PT15.100000078S",
          "durationInSeconds": 15.1,
          "startTime": "PT17.817798612S",
          "startTimeInSeconds": 17.817,
          "trackingEvents": [
            {
              "beaconUrls": [
                "http://exampleadserver.com/tracking?event=impression"
              ],
              "duration": "PT15.100000078S",
              "durationInSeconds": 15.1,
              "eventId": "8104385",
              "eventType": "impression",
              "startTime": "PT17.817798612S",
              "startTimeInSeconds": 17.817
            },
            {
              "beaconUrls": [
                "http://exampleadserver.com/tracking?event=start"
              ],
              "duration": "PT0S",
              "durationInSeconds": 0.0,
              "eventId": "8104385",
              "eventType": "start",
              "startTime": "PT17.817798612S",
              "startTimeInSeconds": 17.817
            },
            {
              "beaconUrls": [
                "http://exampleadserver.com/tracking?event=firstQuartile"
              ],
              "duration": "PT0S",
              "durationInSeconds": 0.0,
```

```

        "eventId": "8104386",
        "eventType": "firstQuartile",
        "startTime": "PT21.592798631S",
        "startTimeInSeconds": 21.592
    },
    {
        "beaconUrls": [
            "http://exampleleadserver.com/tracking?event=midpoint"
        ],
        "duration": "PT0S",
        "durationInSeconds": 0.0,
        "eventId": "8104387",
        "eventType": "midpoint",
        "startTime": "PT25.367798651S",
        "startTimeInSeconds": 25.367
    },
    {
        "beaconUrls": [
            "http://exampleleadserver.com/tracking?event=thirdQuartile"
        ],
        "duration": "PT0S",
        "durationInSeconds": 0.0,
        "eventId": "8104388",
        "eventType": "thirdQuartile",
        "startTime": "PT29.14279867S",
        "startTimeInSeconds": 29.142
    },
    {
        "beaconUrls": [
            "http://exampleleadserver.com/tracking?event=complete"
        ],
        "duration": "PT0S",
        "durationInSeconds": 0.0,
        "eventId": "8104390",
        "eventType": "complete",
        "startTime": "PT32.91779869S",
        "startTimeInSeconds": 32.917
    }
]
}
],
"availId": "8104385",
"duration": "PT15.100000078S",
"durationInSeconds": 15.1,

```



```
    "startTime": "PT17.817798612S",  
    "startTimeInSeconds": 17.817  
  }  
]  
}
```

Hybridmodus mit serverseitigen Ad Beacons

MediaTailor unterstützt einen Hybridmodus für die Sitzungsverfolgung. In diesem Modus sendet der Dienst Werbeverfolgungsereignisse im Zusammenhang mit der Wiedergabe aus, stellt aber die gesamte clientseitige Tracking-Nutzlast für die Sitzung zur Verfügung

Um Hybrid-Tracking mithilfe von Wiedergabepräfixen zu aktivieren, initialisieren Sie vom Player aus eine neue MediaTailor Wiedergabe-Sitzung mit einer Anfrage in einem der folgenden Formate gemäß Ihrem Protokoll:

Example : HLS-Format

```
POST master.m3u8  
{  
  "adsParams": {  
    "deviceType": "ipad"  
  },  
  "reportingMode": "server"  
}
```

Example : DASH-Format

```
POST manifest.mpd  
{  
  "adsParams": {  
    "deviceType": "ipad"  
  },  
  "reportingMode": "server"  
}
```

MediaTailor verwaltet die folgenden Tracking-Ereignisse im Hybridmodus:

- Eindruck
- Starten
- Erstes Quartil

- Mittelpunkt
- Drittes Quartil
- Complete
- `breakStart(vmap)`
- `breakEnd(vmap)`

Kundenseitige Ad-Tracking-Integrationen

In diesem Abschnitt werden Integrationen zwischen MediaTailor und verschiedenen clientseitigen Ad-Tracking-Servern beschrieben.

Themen

- [Öffnen Sie Measurement SDK](#)
- [Kostenloser Datazoom-Player SDKs](#)
- [Roku Advertising Framework \(RAF\)](#)
- [TheoPlayer](#)
- [MediaTailor SDK](#)

Öffnen Sie Measurement SDK

Das Open Measurement SDK (OM SDK) des Interactive Advertising Bureau (IAB) ermöglicht die Messung der Sichtbarkeit und Überprüfung von Anzeigen, die in Web-Video- und nativen App-Umgebungen geschaltet werden.

Bei älteren VAST-Dokumenten der Version 3 sollte der Bestätigungscode mit dem Erweiterungstyp zusammen mit dem Erweiterungsknoten geladen werden. AdVerifications Die Wurzel des Erweiterungsknotens ist ein AdVerifications Knoten mit demselben Schema wie das VAST 4.1-Element.

Um die Einführung des OM SDK zu erleichtern, MediaTailor hat sich das Unternehmen mit Datazoom zusammengetan, um kostenlose Player bereitzustellen, SDKs die für Open Measurement konfiguriert und verifiziert sind. Weitere Informationen finden Sie unter [Kostenloser Datazoom-Player SDKs](#).

Note

MediaTailor unterstützt derzeit nur VAST Version 3.

Example : Bestätigungsknoten in VAST 3, vor Version 4.1

```
...
<Extensions>
  <Extension type="AdVerifications">
    <AdVerifications>
      <Verification vendor="company.com-omid">
        <JavaScriptResource apiFramework="omid" browserOptional="true">
          <![CDATA[https://verification.com/omid_verification.js]]>
        </JavaScriptResource>
        <TrackingEvents>
          <Tracking event="verificationNotExecuted">
            <![CDATA[https://verification.com/trackingurl]]>
          </Tracking>
        </TrackingEvents>
        <VerificationParameters>
          <![CDATA[verification params key/value pairs]]>
        </VerificationParameters>
      </Verification>
    </AdVerifications>
  </Extension>
</Extensions>
```

MediaTailor extrahiert die AdVerifications Daten aus dem <Extensions> Knoten und platziert sie in dem adVerifications Array in der clientseitigen Tracking-Antwort.

Example : ADVerifications-Array in der clientseitigen Tracking-Antwort

```
{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "3062770",
          "adParameters": "",
          "adProgramDateTime": "2023-08-23T16:25:40.914Z",
          "adSystem": "2.0",
          "adTitle": "AD-polarbear-15",
          "adVerifications": [
            {
              "executableResource": [],

```

[illegible]

```

        "beaconUrls": [
            "https://n8ljfs0h09.execute-api.us-west-2.amazonaws.com/v1/impression"
        ],
        "duration": "PT14.982S",
        "durationInSeconds": 14.982,
        "eventId": "3062770",
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT10.11S",
        "startTimeInSeconds": 10.11
    }
],
    "vastAdId": ""
}
],
    "availId": "3062770",
    "availProgramDateTime": "2023-08-23T16:25:40.914Z",
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "meta": null,
    "nonLinearAdsList": [],
    "startTime": "PT10.11S",
    "startTimeInSeconds": 10.11
}
],
    "dashAvailabilityStartTime": null,
    "hlsAnchorMediaSequenceNumber": null,
    "nextToken": "UFQxMC4xMVNfMjAyMy0wOC0yM1QxNjoyNjoyNC4yNDYxMDIxOTBaXzE%3D",
    "nonLinearAvails": []
}

```

Note

Wenden Sie sich an das IAB Tech Lab, um sicherzustellen, dass die Anwendungen jährlich zertifiziert werden, um die Einhaltung der Vorschriften sicherzustellen.

Weitere Informationen zum OM SDK finden Sie unter [Open Measurement SDK](#) auf der Website des IAB Tech Lab.

Kostenloser Datazoom-Player SDKs

Um die Einführung des Players zu erleichtern SDKs, MediaTailor hat es sich mit Datazoom zusammengetan, um kostenlose Player anzubieten SDKs, die mit dem und dem IAB Tech konfiguriert und getestet wurden. [Kundenseitige AWS Elemental MediaTailor Integration mit Google Ad Manager Öffnen Sie Measurement SDK](#)

Das Datazoom Player SDK unterstützt die folgenden Funktionen:

- Live- und VOD-Wiedergabelisten
- DASH- und HLS-Spezifikationen
- Unterstützung von Player-Anbietern für Bitmovin, Exoplayer, Android Media Player, Apple AVPlayer, Brightcove, Chromecast Receiver, Dash.js, hls.js,, Shaka Player, THEO Player, Video.js JWPlayer, Roku und mehr
- IAB Tech Lab Open Measurement-Zertifizierung, sofern für ausgewählte Geräte verfügbar
- Behandlung von Ereignissen per Mausklick
- Dispatcher für Werbeereignisse, z. B. Countdown-Timer für Werbeanzeigen, Anzeigenüberlagerungen und nichtlineare Ereignisse, Beginn und Ende von Werbeunterbrechungen
- Kundenseitiges Werbe-Beaconing
- SDK für die Google Programmatic Access Library (PAL) als optionale Konfigurationseinstellung

Datazoom bietet auch einen kostenpflichtigen Analyse- und Telemetriedienst, den der Spieler unterstützt. SDKs Kunden können die Player-SDK-Telemetrie über die Datazoom-Managementkonsole aktivieren und steuern. [Um auf den Datazoom Player zuzugreifen SDKs und mehr über den Telemetrie- und Analysedienst mit Mehrwert zu erfahren, verwenden Sie die Kontaktinformationen auf der Datazoom-Website.](#)

Roku Advertising Framework (RAF)

Das Roku Ad Framework (RAF) sorgt für ein einheitliches Werberlebnis auf der gesamten Roku-Plattform. Alle Kanäle, einschließlich Videowerbung, müssen die Zertifizierungsanforderungen von Roku für RAF erfüllen. Insbesondere muss die App immer die clientseitige Ereignisauslösung über RAF verwenden. MediaTailor unterstützt als Anbieter für serverseitige Anzeigeneinfügung (SSAI) die clientseitige Auslösung von Ereignissen. Die RAFX SSAI-Adapter bieten Schnittstellen sowohl zu SSAI-Manifestservern oder Stitchern als auch zu RAF. Zu diesen Schnittstellen gehören:

- Analysieren der masterURL Antwort und Extrahieren von playURLAdURL, und Anzeigen-Metadaten.
- Umwandlung von MediaTailor SSAI-Anzeigenmetadaten in RAF-verwendbare Anzeigenmetadaten und Konfiguration von RAF für die Wiedergabe.
- Beobachtung von Stream-Ereignissen und zeitgesteuerten Metadaten.
- Rechtzeitiger Abgleich von Stream-Ereignissen, Anzeigen-Metadaten und Pixeln für das Auslöseereignis
- Pinggen/Abfragen von RAFAdURL, wie vom MediaTailor SSAI-Manifestserver gefordert, anschließendes Parsen und Neukonfigurieren von RAF.

Weitere Informationen zu SSAI-Adaptern für RAF finden Sie unter [Implementieren der serverseitigen Anzeigeneinfügung](#) mithilfe von Roku-Adaptern auf der Roku-Website.

TheoPlayer

TheoPlayer Die Integration mit MediaTailor macht Folgendes:

- Bietet Funktionen zur Unterstützung der MediaTailor clientseitigen Ereignisverfolgung für HLS und DASH sowohl für VOD- als auch für Live-Workflows.
- Unterstützt das Senden von Tracking-Beacons nur für lineare Anzeigen.
- Deaktiviert die Suche während einer Anzeige. Es gibt jedoch keine Logik für das Abspielen einer Anzeige, wenn der Nutzer nach Ablauf der Werbeunterbrechung sucht.

Weitere Informationen zu SSAI und zur Überprüfung im TheoPlayer Internet, Android, iOS und tvOS SDKs finden Sie [MediaTailor](#) auf der TheoPlayer Website. MediaTailor

MediaTailor SDK

AWS Elemental unterhält ein JavaScript basiertes Softwareentwicklungskit (SDK). AWS Elemental stellt das SDK so zur Verfügung, wie es ist, ohne stillschweigende Garantie. Verwenden Sie das SDK als Referenz-Demo, um Ihr Onboarding auf die Nutzung zu optimieren. MediaTailor Das SDK zeigt, wie Sie mit der MediaTailor clientseitigen Tracking-API interagieren. Das SDK implementiert clientseitiges Ad-Tracking und Reporting für aktive Spieler. HTML5 Das SDK initialisiert eine MediaTailor clientseitige Berichtssitzung und fordert dann regelmäßig Informationen zur Anzeigenverfolgung an. Während der Wiedergabe gibt das SDK Anzeigen-Tracking-Ereignisse aus, wenn neue Werbeereignisse erkannt werden.

Das MediaTailor SDK unterstützt die folgenden Funktionen:

- Live- und VOD-Wiedergabelisten
- DASH- und HLS-Spezifikationen
- Behandlung von Ereignissen per Mausklick
- Dispatcher für Werbeereignisse
- Benutzerdefinierte Event-Hooks
- Client-seitiges Werbe-Beaconing. Weitere Informationen zum Senden von Ad Beacons finden Sie unter [Clientseitiges Beaconing](#)

 Note

Reichen Sie ein AWS Support-Ticket ein, um ein JavaScript Beispiel-SDK für zu erhalten MediaTailor. Sie erhalten einen Download-Link für das Paket und seine Dateien.

Durch Werbe-Beacons blättern mit GetTracking

Verwenden Sie den `GetTracking` Endpunkt, um die Anzahl der an einen Player zurückgegebenen Anzeigen einzuschränken. Wenn ein Manifestfenster beispielsweise groß ist und sich über einen langen Zeitraum erstreckt, kann sich die Anzahl der zurückgegebenen Werbe-Beacons auf die Leistung des Players auswirken.

`GetTracking` gibt einen `NextToken` Wert zurück, mit dem Sie die Anzahl der zurückgegebenen Beacons einschränken können, indem Sie die Liste der zurückgegebenen Beacons durchblättern. Sie können die `NextToken` Werte nacheinander durchgehen, um den gewünschten Wert im Feld eines Werbe-Beacons zu finden. `StartTimeInSeconds`

- Beim ersten Aufruf von werden alle möglichen Anzeigen zurückgegeben `GetTracking`, die in das Manifestfenster fallen, einschließlich jeweils eines Werts `NextToken` und.
- Wenn eine `GetTracking` Anfrage kein `A` enthält `NextToken`, werden alle Anzeigen im Manifestfenster zurückgegeben.
- Wenn eine `GetTracking` Anfrage ein enthält, es `NextToken` aber keine neuen Beacons gibt, die MediaTailor zurückgegeben werden können, wird derselbe Wert zurückgegeben `NextToken`, den Sie in der ursprünglichen Anfrage gesendet haben.

- Wenn einer Anzeige keine Beacons mehr entsprechen, GetTracking wird die Anzeige aus der Antwort entfernt.
- Tokens von GetTracking laufen nach 24 Stunden ab. Wenn ein NextToken Wert älter als 24 Stunden ist, gibt der nächste Aufruf von einem GetTracking NextToken Nullwert zurück.

Generalisierte Aufrufsequenz von „vom Spieler“ GetTracking

Eine GetTracking Anfrage vom Client-Spieler ist ein POST mit einem Anforderungstext, der die Werbung NextToken und Beacons im Zusammenhang mit dem Token enthält.

```
https://YouMediaTailorUrl/v1/tracking
{
    "NextToken": "value"
    .
    .
    .
}
```

Die allgemeine Reihenfolge für die Verwendung von GetTracking with NextToken lautet wie folgt:

1. Rufen Sie zum ersten Mal anGetTracking.

Alle Anzeigen und Beacons sowie die ersten NextToken für nachfolgende Aufrufe werden zurückgegeben.

2. Wenn der Wert Null NextToken ist, werden alle Ad Beacons MediaTailor zurückgegeben.
3. Wenn der abgelaufen NextToken ist, MediaTailor wird eine HTTP-Rückgabecode 400-Fehlermeldung zurückgegeben.

Rufen Sie erneut auf, GetTracking um gültige NextToken s abzurufen.

4. Scannen Sie die gesamte Antwort, um herauszufinden, StartTimeInSeconds ob sich ein Werbeanzeigen-Beacon im gewünschten Bereich befindet.
5. Rufen Sie erneut GetTracking mit dem Wert von auf, der dem gewünschten StartTimeInSeconds Wert NextToken zugeordnet ist.
6. Schauen Sie sich bei Bedarf die zurückgegebenen Anzeigen erneut an, bis Sie genau die Anzeigen gefunden haben, die Sie abspielen möchten.

Erweitertes Beispiel

Dieses Beispiel zeigt, wie Sie mithilfe GetTracking von s die Anzahl der NextToken an einen Player zurückgegebenen Werbe-Beacons einschränken können.

MediaTailor erhält eine GetTracking Anfrage. Die Antwort enthält eine Anzeige mit der ID 9935407 und zwei Beacons mit den StartTimeInSeconds Werten 52,286 und 48,332 Sekunden.

MediaTailor sendet die JSON-Antwort wie folgt: NextToken

```
{
  "NextToken": "JF57ITe48t1441mv7TmLKuZLroxDzfIslp6BiSNL1IJmzPVMDN0lqrBYycgMbKEb
  "avails": [
    {
      "ads": [
        {
          "adId": "9935407",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "",
          "creativeSequence": "",
          "duration": "PT15S",
          "durationInSeconds": 15,
          "extensions": [],
          "mediaFiles": {
            "mediaFilesList": [],
            "mezzanine": ""
          },
          "startTime": "PT30S",
          "StartTimeInSeconds": 45,
          "trackingEvents": [
            {
              "beaconUrls": [
                "http://adserver.com/tracking?event=Impression "
              ],
              "duration": "PT0S",
              "durationInSeconds": 0,
              "eventId": "9935414",
              "eventType": "secondQuartile",
              "startTime": "PT52.286S",
              "StartTimeInSeconds": 52.286
            },
            {

```

```

        "beaconUrls": [
            "http://adserver.com/tracking?event=firstQuartile"
        ],
        "duration": "PT0S",
        "durationInSeconds": 0,
        "eventId": "9935412",
        "eventType": "firstQuartile",
        "startTime": "PT48.332S",
        "startTimeInSeconds": 48.332
    }
],
    "vastAdId": ""
}
],
    "startTime": "PT46.47S",
    "startTimeInSeconds": 46.47
}
]
}

```

MediaTailor antwortet auf die nächste GetTracking Anfrage mit dem NextToken Wert: JF57 ITe48t1441mv7 Tm LKu ZLrox Dzflslp 6Bi SNL1 IJmz PVmDN0LQR Mb. BYycg KEb

MediaTailor antwortet mit Anzeigen und Beacons, die denen entsprechen, StartTimeInSeconds die beim vorherigen Anruf gesetzt wurden. NextToken

Gehen Sie davon aus, dass die Antwort jetzt zusätzlich zur vorherigen Anzeige mit der ID 9935407 eine weitere Anzeige mit der ID 9235407 enthält. Die Beacons der Anzeigen-ID 9235407 haben die Werte 132,41 und 70.339. StartTimeInSeconds

MediaTailor durchläuft alle Beacons in der Sitzung, um die Beacons mit StartTimeInSeconds mehr als 52,286 Sekunden auszuwählen, bei denen es sich um Beacon 3 und Beacon 4 aus der Anzeige mit der ID 9235407 handelt:

```

{
    "NextToken": ZkfknbvbsdgsfbsDFRdffg12EdffecFRvhjyjfhdhfnjtsG5SDGN
    "avails": [
        {
            "ads": [
                {
                    "adId": "9235407",

```

```

    "adVerifications": [],
    "companionAds": [],
    "creativeId": "",
    "creativeSequence": "",
    "duration": "PT15.816S",
    "durationInSeconds": 19.716,
    "extensions": [],
    "mediaFiles": {
      "mediaFilesList": [],
      "mezzanine": ""
    },
    "startTime": "PT2M0S",
    "StartTimeInSeconds": 120.0,
    "trackingEvents": [
      {
        "beaconUrls": [
          "http://adserver.com/tracking?event=complete"
        ],
        "duration": "PT0S",
        "durationInSeconds": 0,
        "eventId": "8935414",
        "eventType": "firstQuartile",
        "startTime": "PT1M10.330S",
        "StartTimeInSeconds": 70.339
      },
      {
        "beaconUrls": [
          "http://adserver.com/tracking?event=thirdQuartile"
        ],
        "duration": "PT0S",
        "durationInSeconds": 0,
        "eventId": "8935412",
        "eventType": "secondQuartile",
        "startTime": "PT2M12.41S",
        "StartTimeInSeconds": 132.41
      }
    ],
    "vastAdId": ""
  },
],
"startTime": "PT36.47S",
"StartTimeInSeconds": 36.47
}
]

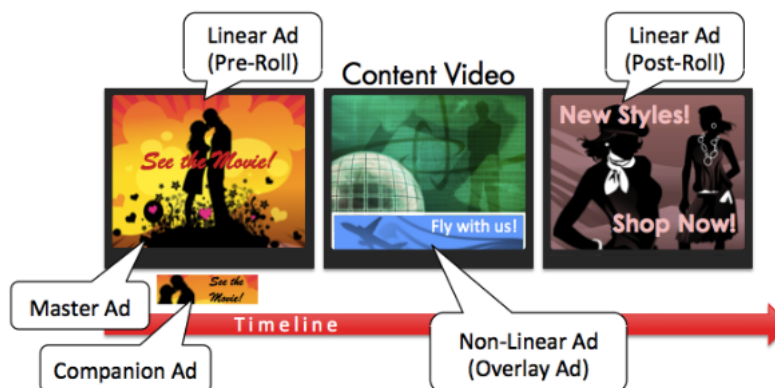
```

```
}
```

Overlay-Anzeigen

Für Live-Streaming-Workflows, bei denen Sie die Monetarisierung erhöhen möchten, ohne das Zuschauererlebnis durch Mid-Roll-Anzeigen zu unterbrechen, können Sie Ihre aktuelle AWS Elemental MediaTailor Integration nutzen, um ein Werbeformat zu entwickeln, das kundenseitig gerendert wird. Diese Art von Werbung wird als Overlay-Werbung bezeichnet. Overlay-Anzeigen sind nichtlineare Videoanzeigen, die in Form von „L-Band-Anzeigen“, „nichtlinearen Videoanzeigen“, „Anzeigen“, „Motion-Overlays“, „picture-in-picture In-Content-Werbung“ oder „Frame-Anzeigen“ geschaltet werden.

MediaTailor erkennt einen SCTE-35-Marker mit Segmentierungstyp als In-Band-Signal für eine Möglichkeit zur Einblendung von Overlay-Werbung. Die SCTE-35-Markierung veranlasst MediaTailor das Senden einer Anfrage an den Ad Decision Server (ADS), der dann mit einer nichtlinearen Anzeigennutzlast in der VAST-Antwort antwortet. MediaTailor analysiert die VAST-Antwort, um das Einfügen von Overlay-Anzeigen zu unterstützen. MediaTailor führt kein Zusammenfügen linearer Anzeigen durch, sondern signalisiert dem Player, dass eine nichtlineare Overlay-Anzeige zum Abspielen verfügbar ist. Diese Signalisierung ermöglicht es dem Player, die nichtlinearen Anzeigen für die Wiedergabe vom clientseitigen Tracking-Endpunkt abzurufen und zu korrelieren. Der Player kümmert sich dann um die Anzeige, die Berichterstattung und andere Aufgaben im Zusammenhang mit diesen Anzeigen. Beispielsweise kann der Entwickler des Players ein Geräte-SDK eines Anbieters verwenden, das Overlay-Anzeigenformate unterstützt. Weitere Informationen zu clientseitigen Tracking-Integrationen finden Sie unter [Kundenseitige Ad-Tracking-Integrationen](#)



Themen

- [Voraussetzungen für die Verwendung von Overlay-Anzeigen mit MediaTailor](#)
- [Erste Schritte mit Overlay-Anzeigen mit MediaTailor](#)
- [Protokollierung und Metriken für Overlay-Anzeigen in MediaTailor](#)
- [Abrechnung für Overlay-Anzeigen in MediaTailor](#)

Voraussetzungen für die Verwendung von Overlay-Anzeigen mit MediaTailor

Die folgenden Voraussetzungen gelten für die Verwendung von Overlay-Anzeigen mit: MediaTailor

- Der Workflow muss live sein, nicht Video-on-Demand (VOD).
- Die Antwort des Ad Decision Servers (ADS) muss so konfiguriert sein, dass in der VAST-Antwort nur nichtlineare Anzeigen zurückgegeben werden. MediaTailor ignoriert alle linearen Anzeigen für das Ad-Stitching.
- Das Manifest muss eine SCTE-35-Zeitsignalnachricht mit Segmentierungstyp verwenden, `id=0x38` um die Overlay-Ad-Funktion aufzurufen.
- Der Streaming-Anbieter muss die Kontrolle über die Anwendung auf dem Client-Gerät haben und in die clientseitige Tracking-API integriert sein. MediaTailor

Erste Schritte mit Overlay-Anzeigen mit MediaTailor

In diesem Abschnitt wird erläutert, wie Sie mit der Verwendung der Overlay-Anzeigenfunktion von beginnen. MediaTailor Sie richten die SCTE-35-Signalisierung ein, konfigurieren die Antworten des Ad Decision Servers (ADS) und richten die Steuerung auf Sitzungsebene ein.

Themen

- [Overlay-Anzeigen aktivieren](#)
- [Verfolgung von Overlay-Anzeigen mit clientseitigen Metadaten](#)

Overlay-Anzeigen aktivieren

MediaTailor Die Unterstützung für Overlay-Anzeigen ist standardmäßig aktiviert. Ein bestimmter SCTE-35-Werbemarkertyp im Manifest löst das Einfügen einer Overlay-Anzeige aus. Da einige Player das clientseitige Rendern von Overlay-Anzeigen möglicherweise nicht unterstützen, können Sie die Funktion auf Sitzungsebene deaktivieren.

So deaktivieren Sie die Unterstützung von Overlay-Anzeigen mithilfe von HLS- oder DASH-Wiedergabepräfixen:

- Initialisieren Sie vom Player aus eine neue MediaTailor Wiedergabesitzung mit einer Anfrage in einem der folgenden Formate, je nach Protokoll:

- Beispiel: HLS-Format

```
GET mediatailorURL/v1/master/hashed-account-id/origin-id/asset-id?  
aws.overlayAvails=off
```

- Beispiel: DASH-Format

```
GET mediatailorURL/v1/master/hashed-account-id/origin-id/asset-id?  
aws.overlayAvails=off
```

So deaktivieren Sie die Unterstützung für Overlay-Ads mithilfe des Sitzungsinitialisierungspräfix:

- Konstruieren Sie auf dem Player einen JSON-Nachrichtentext für die Anfrage zur Sitzungsinitialisierung, um: MediaTailor
 - Um die Ad-Overlay-Unterstützung zu deaktivieren, fügen Sie ein `overlays` Objekt als Schlüssel der obersten Ebene mit einem Wert von `off` hinzu. Der Standardwert `overlays` ist `on`.
 - (Optional) Geben Sie alle Parameter an, die MediaTailor dann innerhalb eines `adsParams` Objekts an das ADS übergeben werden. Diese Parameter entsprechen `[player_params.param]`-Einstellungen in der ADS-Vorlagen-URL der MediaTailor-Konfiguration.

Example HLS:

```
POST master.m3u8  
{  
  "adsParams": {  
    "deviceType": "ipad"  
  },  
  "overlayAvails": "off"  
}
```

Example DASH:

```
POST manifest.mpd
{
  "adsParams": {
    "deviceType": "androidmobile"
  },
  "overlayAvails": "off"
}
```

Manifeste Signalisierung

MediaTailor löst die Unterstützung für Overlay-Ads aus, wenn ein bestimmter SCTE-35-Marker im Manifest erkannt wird. Das erforderliche Signal ist ein Splice-Befehl vom Typ 6 oder ein Zeitsignal, bei dem es sich um ein Startsignal für Provider Overlay Advertisement handelt. Dieses Signal hat die Segmentierungstyp-ID von 0x38

Das folgende Beispiel zeigt den 0x38 SCTE-35-Marker in einem JSON-Objekt.

```
{
  "tableId": 252,
  "selectionSyntaxIndicator": false,
  "privateIndicator": false,
  "sectionLength": 53,
  "protocolVersion": 0,
  "encryptedPacket": false,
  "encryptedAlgorithm": 0,
  "ptsAdjustment": 0,
  "cwIndex": 0,
  "tier": 4095,
  "spliceCommandLength": 5,
  "spliceCommandType": 6,
  "spliceCommand": {
    "specified": true,
    "pts": 1800392
  },
  "descriptorLoopLength": 31,
  "descriptors": [
    {
      "spliceDescriptorTag": 2,
      "descriptorLength": 29,
      "identifier": "CUEI",
      "segmentationEventId": 158389361,
```



```

"segmentationEventCancelIndicator": false,
"programSegmentationFlag": true,
"segmentationDurationFlag": true,
"deliveryNotRestrictedFlag": false,
"webDeliveryAllowedFlag": true,
"noRegionalBlackoutFlag": true,
"archiveAllowedFlag": true,
"deviceResctrictions": 3,
"segmentationDuration": 1350000,
"segmentationUpidType": 9,
"segmentationUpidLength": 7,
"segmentationUpid": {
  "0": 111,
  "1": 118,
  "2": 101,
  "3": 114,
  "4": 108,
  "5": 97,
  "6": 121
},
"segmentationTypeId": 56,
"segmentNum": 1,
"segmentsExpected": 0
}
],
"crc": 2510422713
}

```

Das folgende Beispiel zeigt, dass das SCTE-35-Signal als binärer Wert (Basis 32/Hexadezimal) dargestellt wird:

```
0xfc303500000000000000ff00506fe001b78c8001f021d435545490970d4717fdf00000dbba009076f7665726c617
```

Das folgende Beispiel zeigt den SCTE-35-Marker sowohl in HLS- als auch in DASH-Manifesten.

Example : HLS-Manifest

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:419
#EXT-X-DISCONTINUITY-SEQUENCE:3
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:09.231Z
```

```
#EXTINF:6.02,  
https://aws.cloudfront.net/media/asset1/index1_00007.ts  
#EXT-X-DISCONTINUITY  
#EXT-X-KEY:METHOD=NONE  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:15.251Z  
#EXTINF:6.0,  
https://aws.cloudfront.net/media/asset1/index1_00001.ts  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:21.251Z  
#EXTINF:4.0,  
https://aws.cloudfront.net/media/asset1/index1_00002.ts  
#EXT-X-DISCONTINUITY  
#EXT-X-DATERANGE:ID="1692073825251-30-1", START-  
DATE="2023-08-15T04:30:25.251Z", DURATION=10.0, PLANNED-DURATION=10.0, SCTE35-  
OUT=0xfc303500000000000000ffff00506fe001b78c8001f021d435545490970d4717fdf00000dbba009076f7665726  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:25.251Z  
#EXTINF:2.0,  
https://aws.cloudfront.net/media/asset1/index1_00003.ts  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:27.251Z  
#EXTINF:6.0,  
https://aws.cloudfront.net/media/asset1/index1_00004.ts  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:33.251Z  
#EXTINF:2.0,  
https://aws.cloudfront.net/media/asset1/index1_00005.ts  
#EXT-X-DISCONTINUITY  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:35.251Z  
#EXTINF:4.0,  
https://aws.cloudfront.net/media/asset1/index1_00006.ts  
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:39.251Z  
#EXTINF:6.02,  
https://aws.cloudfront.net/media/asset1/index1_00007.ts
```

Example : DASH-Manifest

```
<?xml version="1.0"?>
<MPD xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  availabilityStartTime="2023-08-15T16:34:05.911Z" minBufferTime="PT30S"
  minimumUpdatePeriod="PT2S" profiles="urn:mpeg:dash:profile:isoff-live:2011"
  publishTime="2023-08-15T16:34:17.950Z" suggestedPresentationDelay="PT20S"
  timeShiftBufferDepth="PT1M30S" type="dynamic"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/ittf/
PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd">
  <Period xmlns="urn:mpeg:dash:schema:mpd:2011" id="1692117245944_1" start="PT0.033S">
```

```

<BaseUrl>https://aws.cloudfront.net/out/v1/abc/123/def/</BaseUrl>
<EventStream schemeIdUri="urn:scte:scte35:2013:xml" timescale="90000">
  <Event duration="900000">
    <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="0" tier="4095">
      <scte35:TimeSignal>
        <scte35:SpliceTime ptsTime="0"/>
      </scte35:TimeSignal>
      <scte35:SegmentationDescriptor segmentNum="0" segmentationDuration="900000"
segmentationEventCancelIndicator="false" segmentationEventId="1"
segmentationTypeId="56" segmentsExpected="0" subSegmentNum="0"
subSegmentsExpected="0">
        <scte35:SegmentationUpid segmentationUpidFormat="hexBinary"
segmentationUpidType="14">63736f7665726c6179</scte35:SegmentationUpid>
      </scte35:SegmentationDescriptor>
    </scte35:SpliceInfoSection>
  </Event>
</EventStream>
<AdaptationSet bitstreamSwitching="true" mimeType="video/mp4"
segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
  <Representation bandwidth="3000000" codecs="avc1.4D4028" frameRate="30/1"
height="1080" id="1" width="1920">
    <SegmentTemplate initialization="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/
index_video_1_0_init.mp4" media="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/index_video_1_0_
$Number$.mp4" presentationTimeOffset="0" startNumber="1" timescale="30000">
      <SegmentTimeline>
        <S d="60000" r="6" t="1000"/>
        <S d="30000" t="421000"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>
  <Representation bandwidth="2499968" codecs="avc1.4D4028" frameRate="30/1"
height="1080" id="2" width="1920">
    <SegmentTemplate initialization="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/
index_video_2_0_init.mp4" media="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/index_video_2_0_
$Number$.mp4" presentationTimeOffset="0" startNumber="1" timescale="30000">
      <SegmentTimeline>
        <S d="60000" r="6" t="1000"/>
        <S d="30000" t="421000"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>

```

```

        </SegmentTemplate>
    </Representation>
    <Representation bandwidth="2200000" codecs="avc1.4D401F" frameRate="30/1"
height="720" id="3" width="1280">
        <SegmentTemplate initialization="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/
index_video_3_0_init.mp4" media="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/index_video_3_0_
$Number$.mp4" presentationTimeOffset="0" startNumber="1" timescale="30000">
            <SegmentTimeline>
                <S d="60000" r="6" t="1000"/>
                <S d="30000" t="421000"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <Label>Alternate Audio</Label>
    <Representation audioSamplingRate="48000" bandwidth="128000" codecs="mp4a.40.2"
id="9">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        <SegmentTemplate initialization="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/
index_audio_9_0_init.mp4" media="../
cf684d31ec9e451ca98d2349989f6c0a/855c733eed20493ab3cc1100750bcf0b/index_audio_9_0_
$Number$.mp4" presentationTimeOffset="0" startNumber="1" timescale="48000">
            <SegmentTimeline>
                <S d="98304" t="0"/>
                <S d="96256" t="98304"/>
                <S d="95232" t="194560"/>
                <S d="96256" r="2" t="289792"/>
                <S d="95232" t="578560"/>
                <S d="46080" t="673792"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
</AdaptationSet>
</Period>
</MPD>

```

Antwort des Ad Decision Servers (ADS)

Die ADS-Antwort muss ein gültiges Tracking-Ereignis enthalten. Bei dem Tracking-Ereignis kann es sich mindestens um ein Impression Tracking-Ereignis handeln. Das Tracking-Event sollte mindestens eine NonLinear Anzeige enthalten. Bei dieser Anzeige handelt es sich um eine Overlay-Anzeige in Form einer statischen, HTML- oder iFrame-Ressource.

```
<vmap AdBreak breaktype="linear" breakId="csoverlay"
```

Wenn es sich bei der VAST-Antwort um eine VMAP mit `breakType` of `nonlinear`, befinden sich die verfügbaren Metadaten im Stammobjekt. `nonLinearAvails` Wenn es sich bei der VAST-Antwort um eine VMAP mit einem `breakType` of `linear` oder um eine einfache VAST-Antwort ohne VMAP handelt, befinden sich die Avail-Metadaten innerhalb des Stammobjekts. `avails`

Die folgende VAST-Antwort ist eine umschlossene VMAP-Antwort mit dem Wert. `breakType` `linear`

Unterstützt neben der umschlossenen VMAP-Antwort MediaTailor auch eine umschlossene VMAP-Antwort mit einem `breakType` Wert von `nonlinear` und eine einfache VAST-Antwort.

```
<?xml version="1.0" encoding="utf-8"?>
<vmap:VMAP xmlns:vmap="http://www.iab.net/vmap-1.0" version="1.0">
  <vmap:AdBreak breakType="linear" breakId="csoverlay">
    <vmap:AdSource allowMultipleAds="true" followRedirects="true" id="1">
      <vmap:VASTAdData>
        <VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="3.0"
xsi:noNamespaceSchemaLocation="vast.xsd">
          <Ad sequence="1">
            <InLine>
              <AdSystem>2.0</AdSystem>
              <AdTitle>2</AdTitle>
              <Impression><![CDATA[https://adserver.com/beacon=impression]]></
Impression>
              <Creatives>
                <Creative>
                  <NonLinearAds>
                    <NonLinear width="640" height="360" id="18">
                      <StaticResource creativeType="text/js_ref"><![CDATA[https://
client-side-ads.com/tags/static/ctv-generic/overlay001.json?iv\_geo\_country%3DUS%26]]></
StaticResource>
                    </NonLinear>
```

```

        </NonLinearAds>
      </Creative>
    </Creatives>
  </Inline>
</Ad>
</VAST>
</vmap:VASTAdData>
</vmap:AdSource>
<vmap:TrackingEvents>
  <vmap:Tracking event="breakStart"><![CDATA[https://adserver.com/
beacon=breakstartimpression]]></vmap:Tracking>
  <vmap:Tracking event="breakEnd"><![CDATA[https://adserver.com/
beacon=breakendimpression]]></vmap:Tracking>
</vmap:TrackingEvents>
</vmap:AdBreak>
</vmap:VMAP>

```

Example 1: Quelle des DASH-Manifests für MediaTailor

```

<?xml version="1.0" encoding="utf-8"?>
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd" id="201"
  type="dynamic" publishTime="2022-11-07T19:59:05+00:00" minimumUpdatePeriod="PT2S"
  availabilityStartTime="2022-11-07T06:57:11.250000+00:00" minBufferTime="PT10S"
  suggestedPresentationDelay="PT20.000S" timeShiftBufferDepth="PT58.999S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011">
  <Period start="PT46827.601S" id="0" duration="PT88.321S">
    ...
  </Period>
  <Period start="PT46915.922S" id="45" duration="PT6.006S">
    <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2014:xml+bin">
      <Event duration="540000" id="144">
        <scte35:Signal>
          <scte35:Binary>SCTE35-binary</scte35:Binary>
        </scte35:Signal>
      </Event>
    </EventStream>
    ...
  </Period>
  <Period start="PT46921.928S" id="49">
    ...

```

```
</Period>
</MPD>
```

Example 2: MediaTailor personalisiertes DASH-Manifest, das eine Werbe-ID-Dekoration enthält

```
<?xml version="1.0" encoding="utf-8"?>
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd" id="201"
  type="dynamic" publishTime="2022-11-07T19:59:05+00:00" minimumUpdatePeriod="PT2S"
  availabilityStartTime="2022-11-07T06:57:11.250000+00:00" minBufferTime="PT10S"
  suggestedPresentationDelay="PT20.000S" timeShiftBufferDepth="PT58.999S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011">
  <Period start="PT46827.601S" id="0" duration="PT88.321S">
    ...
  </Period>
  <Period start="PT46915.922S" id="45" duration="PT6.006S">
    <EventStream schemeIdUri="urn:sva:advertising-wg:ad-id-signaling" timescale="90000">
      <Event presentationTime="13500000" duration="1351350">
        <![CDATA[{"version": 1,"identifiers": [{"scheme":
"urn:smp:ul:060E2B34.01040101.01200900.00000000","value": "adId","ad_position":
"adId", "ad_type":"overlay","creative_id": "creativeId","tracking_uri":
"trackingUri"}]]]></Event>
      </EventStream>
      ...
    </Period>
    <Period start="PT46921.928S" id="49">
      ...
    </Period>
  </MPD>
```

Verfolgung von Overlay-Anzeigen mit clientseitigen Metadaten

MediaTailor platziert die Overlay-Anzeigen ohne Erfolg `nonLinearAdsList`. Die MediaTailor clientseitige Tracking-API hat zwei Stammobjekte, die als und bezeichnet werden. `avails` `nonLinearAvails` Wenn es sich bei der VAST-Antwort um eine VMAP mit `breakType` of `handelt nonlinear`, befinden sich die verfügbaren Metadaten innerhalb des Stammobjekts. `nonLinearAvails` Wenn es sich bei der VAST-Antwort um eine VMAP mit einem `breakType` of `linear` oder um eine einfache VAST-Antwort ohne VMAP handelt, befinden sich die Avail-Metadaten innerhalb des Stammobjekts. `avails`

Weitere Informationen zur clientseitigen Nachverfolgung finden Sie unter. [Kundenseitiges Ad-Tracking](#)

Das folgende Beispiel zeigt eine einfache VAST-Antwort oder VMAP-Antwort mit dem Wert. `breakType linear`

```
{
  "avails": [
    {
      "adBreakTrackingEvents": [
        {
          "beaconUrls": [
            "https://adserver.com/beacon=breakstartimpression"
          ],
          "eventType": "breakStart"
        },
        {
          "beaconUrls": [
            "https://adserver.com/beacon=breakendimpression"
          ],
          "eventType": "breakEnd"
        }
      ],
      "adMarkerDuration": null,
      "ads": [],
      "availId": "828",
      "availProgramDateTime": null,
      "duration": "PT0S",
      "durationInSeconds": 0,
      "meta": null,
      "nonLinearAdsList": [
        {
          "extensions": null,
          "nonLinearAdList": [
            {
              "adId": "",
              "adParameters": null,
              "adSystem": "2.0",
              "adTitle": "2",
              "apiFramework": null,
              "clickThrough": null,
              "clickTracking": null,
              "clickTrackingId": null,
            }
          ]
        }
      ]
    }
  ]
}
```



```

        "creativeAdId": "",
        "creativeId": "18",
        "creativeSequence": "",
        "duration": null,
        "durationInSeconds": 0,
        "expandedHeight": null,
        "expandedWidth": null,
        "height": "360",
        "htmlResource": null,
        "iFrameResource": null,
        "maintainAspectRatio": false,
        "minSuggestedDuration": null,
        "scalable": false,
        "staticResource": "https://client-side-ads.com/tags/static/ctv-generic/
overlay001.json?iv_geo_country%3DUS%26",
        "staticResourceCreativeType": "text/js_ref",
        "width": "640"
    }
],
"trackingEvents": [
    {
        "beaconUrls": [
            "https://adserver.com/beacon=impression"
        ],
        "duration": null,
        "durationInSeconds": 0,
        "eventId": null,
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": null,
        "startTimeInSeconds": 0
    }
]
}
],
"startTime": "PT1M46.08S",
"startTimeInSeconds": 106.08
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": null,
"nonLinearAvails": []

```

```
}
```

Das folgende Beispiel zeigt eine einfache VMAP-Antwort mit dem `breakType` Wert. `nonlinear`

```
{
  "avails": [],
  "dashAvailabilityStartTime": null,
  "hlsAnchorMediaSequenceNumber": null,
  "nextToken": null,
  "nonLinearAvails": [
    {
      "adBreakTrackingEvents": [
        {
          "beaconUrls": [
            "https://adserver.com/beacon=breakstartimpression"
          ],
          "eventType": "breakStart"
        },
        {
          "beaconUrls": [
            "https://adserver.com/beacon=breakendimpression"
          ],
          "eventType": "breakEnd"
        }
      ],
      "adMarkerDuration": null,
      "ads": [],
      "availId": "828",
      "availProgramDateTime": null,
      "duration": "PT0S",
      "durationInSeconds": 0,
      "meta": null,
      "nonLinearAdsList": [
        {
          "extensions": null,
          "nonLinearAdList": [
            {
              "adId": "",
              "adParameters": null,
              "adSystem": "2.0",
              "adTitle": "2",
              "apiFramework": null,
              "clickThrough": null,

```

```

        "clickTracking": null,
        "clickTrackingId": null,
        "creativeAdId": "",
        "creativeId": "18",
        "creativeSequence": "",
        "duration": null,
        "durationInSeconds": 0,
        "expandedHeight": null,
        "expandedWidth": null,
        "height": "360",
        "htmlResource": null,
        "iFrameResource": null,
        "maintainAspectRatio": false,
        "minSuggestedDuration": null,
        "scalable": false,
        "staticResource": "https://client-side-ads.com/tags/static/ctv-generic/  
overlay001.json?iv_geo_country%3DUS%26",
        "staticResourceCreativeType": "text/js_ref",
        "width": "640"
    },
    ],
    "trackingEvents": [
        {
            "beaconUrls": [
                "https://adserver.com/beacon=impression"
            ],
            "duration": null,
            "durationInSeconds": 0,
            "eventId": null,
            "eventProgramDateTime": null,
            "eventType": "impression",
            "startTime": null,
            "startTimeInSeconds": 0
        }
    ]
},
],
"startTime": "PT1M46.08S",
"startTimeInSeconds": 106.08
}
]
}

```

Protokollierung und Metriken für Overlay-Anzeigen in MediaTailor

In diesem Abschnitt werden die Protokollierung und die Messwerte für Overlay-Anzeigen erläutert. MediaTailor Weitere Informationen zum Einrichten der Protokollierung finden Sie unter [Ressourcen überwachen und kennzeichnen AWS Elemental MediaTailor](#).

Themen

- [CloudWatch Logs](#)
- [CloudWatch-Metriken](#)

CloudWatch Logs

CloudWatch sammelt die folgenden Protokollinformationen zu Overlay-Anzeigen:

- VAST_RESPONSE- Zeigt Informationen zur Liste der nichtlinearen Anzeigen an.
- FILLED_PROVIDER_OVERLAY- Zeigt Informationen zu den nichtlinearen Anzeigen an.

Note

Das RAW_ADS_RESPONSE ist ein optionales Ereignis, das die ursprüngliche Antwort des ADS anzeigt. Die Verwendung dieses Ereignisses ist besonders in einer Staging- und Testumgebung hilfreich. Um dieses Ereignis für eine Konfiguration oder ein Konto zu aktivieren, reichen Sie ein Ticket an den AWS Support ein.

CloudWatch-Metriken

MediaTailor erfasst Messwerte für Overlay-Anzeigen getrennt von anderen ADS-Metriken. MediaTailor sammelt diese Messwerte, nachdem die Anzeigen erfolgreich aus dem ADS abgerufen wurden. Sie müssen die GetTracking API nicht abfragen, um die Metriken zu sammeln.

In der folgenden Tabelle werden CloudWatch Metriken für Overlay-Anzeigen beschrieben:

Metrik	Beschreibung
AdDecisionServer.OverlayAds	Die Anzahl der Overlay-Anzeigen, die innerhalb des von Ihnen angegebenen CloudWatch

Metrik	Beschreibung
	Zeitraums in den ADS-Antworten enthalten waren.
<code>AdDecisionServer.OverlayErrors</code>	Die Anzahl der Antworten ohne 200 HTTP-Statuscode, der leeren Antworten und der Antworten mit Timeout, die innerhalb des von Ihnen angegebenen CloudWatch Zeitraums vom ADS MediaTailor empfangen wurden.
<code>AdDecisionServer.OverlayFilled</code>	Die Anzahl der Avails, die erfolgreich mit mindestens einer Overlay-Anzeige gefüllt wurden: • 1 — Es gibt mindestens eine gültige Anzeige. • 0 — Entweder wurden MediaTailor keine Overlay-Anzeigen angezeigt, oder es ist ein anderer Fehler aufgetreten. <code>SampleCount</code> verfolgt die Anzahl der ausgefüllten Avails. <code>Sum</code> verfolgt die Anzahl der erfolgreich ausgefüllten Overlay-Avails.
<code>AdDecisionServer.OverlayMinSuggestedDuration</code>	Die Summe der <code>minSuggestedDuration</code> Dauer aller Anzeigen in Millisekunden, die innerhalb des von Ihnen angegebenen MediaTailor Zeitraums vom ADS empfangen wurden. CloudWatch Wenn <code>minSuggestedDuration</code> nicht angegeben, entspricht die angezeigte Dauer der geplanten Dauer.
<code>AdDecisionServer.OverlayLatency</code>	Die Antwortzeit in Millisekunden für Anfragen an das MediaTailor ADS.

Metrik	Beschreibung
<code>AdDecisionServer.OverlayTimeouts</code>	Die Anzahl der Anfragen an das ADS mit Timeout in dem von Ihnen angegebenen CloudWatch Zeitraum.
<code>AdsBilled</code>	Weitere Informationen zu fakturierten Anzeigen finden Sie unter. Abrechnung für Overlay-Anzeigen in MediaTailor
<code>Avail.*</code>	Weil MediaTailor keine Planung für Overlay-Anzeigen durchgeführt wird und CloudWatch keine Messwerte angezeigt werden <code>Avail.X</code> .
<code>SkippedReason.*</code>	Weil MediaTailor es keine Planung für Overlay-Anzeigen durchführt, CloudWatch keine Metriken anzeigt. <code>SkippedReason.X</code>

Abrechnung für Overlay-Anzeigen in MediaTailor

MediaTailor stellt Kunden auf der Grundlage der Anzahl der nichtlinearen Anzeigen in der ADS-Antwort eine Rechnung. In dieser Zahl sind auch nichtlineare Anzeigen enthalten, die über die Dauer der Pause hinausgehen. Sobald der MediaTailor Vorrat aufgefüllt ist, werden die ausgefüllten Anzeigen in Rechnung gestellt.

Bei Prefetch-Workflows werden Anzeigen MediaTailor nicht beim Abrufen des Prefetchs in Rechnung gestellt, sondern erst, wenn im Nutzungsfenster für diese Sitzung eine kompatible Anzeige verfügbar ist.

Weitere Abrechnungsinformationen finden Sie unter. <https://aws.amazon.com/mediatailor/pricing/>

Werbe-ID-Dekoration

AWS Elemental MediaTailor führt beim Übergang von Inhalten zu Werbeunterbrechungen serverseitiges Ad-Stitching durch. MediaTailor kann das Manifest mit Metadaten zu den zusammengestellten Anzeigen konditionieren. Dies kann die folgenden Vorteile bieten:

- Die Videostartzeit (VST) verbessert sich

- MediaTailor kann ein Hybridmodell aus serverseitiger und clientseitiger Anzeigeneinfügung unterstützen
- In serverseitigen Sitzungen können Zeitlinien für die Wiedergabe mithilfe von Positionsmarkierungen für Anzeigen erstellt werden
- Für clientseitige Sitzungen, für die bereits Wiedergabezeitpläne mit der MediaTailor API erstellt wurden, verbessert sich das Sitzungs-VST, da die Sitzung nicht auf den Aufruf der Tracking-API angewiesen ist, um die Zeitleiste zu erstellen
- Es ist möglich, sowohl serverseitige Anzeigen MediaTailor einzufügen als auch clientseitig gerenderte Anzeigen, die in der Szene angezeigt werden, zu nutzen. Auf diese Weise benötigt das Software Development Kit (SDK) eines Players keine separate Integration, um Ad-Serving-Entitäten direkt für clientseitige Werbung aufzurufen. MediaTailor kann die Anzeigen über das Manifest und die clientseitige Tracking-API verkaufen.

Es gibt Standards, nach denen jedem kreativen Anzeigen-Asset eine eindeutige Kennung zugewiesen wird. Diese Zuordnung ermöglicht es Werbetreibenden, Agenturen, Anbietern und Publishern, ein kreatives Anzeigen-Asset in ihren unabhängigen Workflows miteinander zu verknüpfen. Da sich die Kennzahlen und die Überwachung von Streams ständig verbessern und immer mehr Vertriebspartner serverbasierte Insertion-Architekturen einsetzen, besteht die Notwendigkeit, die den einzelnen kreativen Assets zugewiesenen Identifikatoren innerhalb einer verschachtelten/zusammengefügt Präsentation, wie z. B. innerhalb des personalisierten Manifests, präzise zu kommunizieren.

Themen

- [Aktivierung der Signalisierung von Werbe-IDs für Sitzungen](#)
- [Manifeste und Einfügen von Anzeigenmetadaten](#)
- [Interaktionen mit dem Ad Decision Server \(ADS\)](#)
- [Client-seitige Tracking-API](#)

Aktivierung der Signalisierung von Werbe-IDs für Sitzungen

Die Signalisierungsfunktion für die Anzeigen-ID muss während der Sitzungsinitialisierung aktiviert sein. Das Verfahren zur Aktivierung der Funktion unterscheidet sich von der Erstellung von Sitzungen mit dem HLS/DASH-Wiedergabeprefix (implizite Sitzungsinitialisierung) und dem Sitzungsinitialisierungsprefix (explizite Sitzungsinitialisierung).

So aktivieren Sie die Anzeigen-ID für die Sitzung mithilfe von HLS/DASH-Wiedergabepräfixen

- Initialisieren Sie vom Player aus eine neue MediaTailor Wiedergabesitzung mithilfe einer Anfrage in einem der folgenden Formate gemäß Ihrem Protokoll:
- Beispiel: HLS-Format

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/<asset-id>?  
aws.adSignalingEnabled=true
```

- Beispiel: DASH-Format

```
GET <mediatailorURL>/v1/dash/<hashed-account-id>/<origin-id>/<asset-id>?  
aws.adSignalingEnabled=true
```

So aktivieren Sie die Anzeigen-ID für die Sitzung mithilfe des Sitzungsinitialisierungspräfixes

- Erstellen Sie auf dem Player einen JSON-Nachrichtentext für die Anfrage zur Sitzungsinitialisierung, um: MediaTailor
- Geben Sie innerhalb eines adsParams Objekts alle Parameter an, die an das ADS übergeben werden MediaTailor sollen. Diese Parameter entsprechen den [player_params.param] Einstellungen in der ADS-Vorlagen-URL der MediaTailor Konfiguration.
- Um die Signalisierung der Anzeigen-ID zu aktivieren, fügen Sie ein adSignaling Objekt als Objekt der obersten Ebene hinzu und fügen Sie darin einen Parameter mit dem Namen enabled und dem Wert von true hinzu. Der adSignaling Standardwert ist disabled.
- Beispiel: HLS-Format

```
POST master.m3u8  
{  
  "adsParams": {  
    "deviceType": "ipad"  
  },  
  "adSignaling": {  
    "enabled": "true"  
  }  
}
```


- Beispiel: DASH-Format

```
POST manifest.mpd
{
  "adsParams": {
    "deviceType": "ipad"
  },
  "adSignaling": {
    "enabled": "true"
  }
}
```

Manifeste und Einfügen von Anzeigenmetadaten

MediaTailor fügt dem Manifest beim Zusammenfügen von Anzeigen die eindeutige ID hinzu, die jedem zusammengestellten Werbemittel zugeordnet ist. MediaTailor ermittelt die eindeutige ID des Werbemittels anhand des `id` Attributwerts dieses Werbemittels in der VAST-Antwort. Wenn dem Creative ein ID-Attributwert fehlt, veröffentlicht MediaTailor es einen leeren Wert (`id=""`).

MediaTailor verwendet ein In-Manifest-Metadaten-Signal, um die Abhängigkeiten zwischen der Client-Tracking-API für Anzeigenkreativ-Metadaten und dem Timing/der Positionierung innerhalb des Gesamtzeitplans zu entkoppeln. Diese Entkopplung reduziert die Wiedergabe-Latenz (insbesondere in VOD-Szenarien), bei denen die Benutzeroberfläche (UI) des Players die Positionen der Werbeunterbrechungen in der Timeline rendert, bevor die Wiedergabe initialisiert wird.

Die hinzugefügten Metadaten haben die folgenden Formen:

- Bei HLS-Manifesten haben die hinzugefügten Metadaten die Form von DATERANGE Tags für jede Anzeige im Verfügbarkeitszeitraum.
- Bei DASH-Manifesten haben die hinzugefügten Metadaten die Form eines Event Elements innerhalb jedes Anzeigenzeitraums.

Der folgende JSON-Nachrichtentext zeigt ein Beispiel für eine VAST-Antwort:

```
{
  "version": 1,
  "identifiers": [
    {
```

```

    "scheme": "urn:smp:ul:060E2B34.01040101.01200900.00000000",
    "value": "creativeId",
    "ad_position": "adId",
    "ad_type": "adType",
    "tracking_uri": "trackingUri",
    "custom_vast_data": "customVastData"
  }
]
}

```

Für das obige Beispiel gilt:

- *creativeId* ist der Id Attributwert des Creative Elements für die Anzeige
- *adId* ist entweder die HLS-Sequenznummer, die dem Beginn der Anzeige zugeordnet ist, oder die DASH-Perioden-ID der Anzeige
- *adType* ist entweder `avail` oder `overlay`, basierend auf der VAST-Antwort
- *trackingUri* ist der relative Tracking-Endpoint für die MediaTailor Sitzung im folgenden Format `../../../../tracking/hashed-account-id/origin-id/session-id`
- *customVastData* ist ein Wert, der aus der `creative_signaling` VAST-Erweiterung MediaTailor extrahiert wird. MediaTailor verwendet den Inhalt des CDATA-Knotens, falls vorhanden. Weitere Informationen und ein Beispiel für eine VAST-Antwort finden Sie im [Interaktionen mit dem Ad Decision Server \(ADS\)](#) Abschnitt.

Personalisierung von HLS-Manifesten mit Anzeigenmetadaten

Fügt bei einem HLS-Live-Stream MediaTailor nur Metadaten hinzu, wenn der Stream PROGRAM-DATE-TIME Tags enthält, und zwar mindestens einmal pro Manifestdauer. MediaTailor fügt PROGRAM-DATE-TIME bei einem Video-on-Demand-Stream (VOD) mindestens ein Segment im personalisierten Manifest hinzu, wobei die Startzeit für jedes VOD-Asset Epoche Null () ist. 1970-01-01T00:00:00Z Wenn das Ursprungsmanifest bereits PROGRAM-DATE-TIME Inhalt enthält, wird dieser Inhalt beibehalten MediaTailor .

MediaTailor personalisiert das Manifest mit den vom Ad Decision Server (ADS) zurückgegebenen Creatives. Enthält für jede Anzeige MediaTailor außerdem ein DATERANGE Tag, das sich über die gesamte Laufzeit der Anzeige erstreckt. Das DATERANGE Tag-Format ähnelt dem Format, das im Abschnitt [Ad Creative Signaling in DASH und HLS](#) in der Version 2023 der technischen Publikation von SVA beschrieben wird.

Das DATERANGE, was MediaTailor generiert wird, hat eindeutige ID-Werte. Um die Eindeutigkeit zu gewährleisten (unter Berücksichtigung der in [Mapping SCTE-35 into](#) angegebenen Richtlinien EXT-X-DATERANGE), MediaTailor wird die MEDIA-SEQUENCE Nummer des ersten Anzeigensegments der Avail mit der Sequenznummer der Anzeige innerhalb des Avails verknüpft.

Bei unzureichend ausgefüllten Werbeunterbrechungen in Konfigurationen, bei denen Slate aktiviert ist, werden die Slate-Segmente an das Ende des Avails MediaTailor angehängt, getrennt durch ein Tag, aber ohne Metadaten. DISCONTINUITY DATERANGE

Fügt für jede Anzeige, die in das personalisierte Manifest eingebunden wird, die kreativen Metadaten MediaTailor hinzu, die als Base64-kodierte Daten in einem benutzerdefinierten Tag dargestellt werden. DATERANGE

Example Linearer HLS-Ursprung (): #EXT-X-CUE-OUT

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:398
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:20:01.397Z
#EXTINF:6.006,
index_1_398.ts?m=1676054627
#EXTINF:5.873,
index_1_399.ts?m=1676054627
#EXT-OATCLS-SCTE35:/DA1AAAAAyYAP/wFAUAAAACf+//jP197P4AUmNiAAEBAQAase4/gA==
#EXT-X-CUE-OUT:59.993
#EXTINF:6.139,
index_1_400.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=6.139,Duration=59.993,SCTE35=/DA1AAAAAyYAP/wFAUAAAACf+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_401.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=12.145,Duration=59.993,SCTE35=/DA1AAAAAyYAP/wFAUAAAACf+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_402.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=18.151,Duration=59.993,SCTE35=/DA1AAAAAyYAP/wFAUAAAACf+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_403.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=24.157,Duration=59.993,SCTE35=/DA1AAAAAyYAP/wFAUAAAACf+//jP197P4AUmNiAAEBAQAase4/gA==
```

```
#EXTINF:6.006,
index_1_404.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=30.163,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf
+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_405.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=36.169,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf
+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_406.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=42.175,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf
+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_407.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=48.181,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf
+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_408.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=54.187,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf
+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:5.806,
index_1_409.ts?m=1676054627
#EXT-X-CUE-IN
#EXTINF:6.206,
index_1_410.ts?m=1676054627
#EXTINF:6.006,
index_1_411.ts?m=1676054627
#EXTINF:6.006,
index_1_412.ts?m=1676054627
```

Example Linearer HLS-Ursprung ()#EXT-X-DATERANGE:

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:25
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:53.389Z
#EXTINF:6.006,
index_1_25.ts?m=1676056675
#EXTINF:6.006,
index_1_26.ts?m=1676056675
#EXTINF:6.006,
index_1_27.ts?m=1676056675
```

```
#EXTINF:1.869,
index_1_28.ts?m=1676056675
#EXT-X-DATERANGE:ID="2",START-DATE="2023-02-10T19:20:13.276Z",PLANNED-
DURATION=59.993,SCTE35-
OUT=0xFC302500000003289800FFF01405000000027FEFFF8CF97DECFE00526362000101010000B1EE3F80
#EXTINF:6.139,
index_1_29.ts?m=1676056675
#EXTINF:6.006,
index_1_30.ts?m=1676056675
#EXTINF:6.006,
index_1_31.ts?m=1676056675
#EXTINF:6.006,
index_1_32.ts?m=1676056675
#EXTINF:6.006,
index_1_33.ts?m=1676056675
#EXTINF:6.006,
index_1_34.ts?m=1676056675
#EXTINF:6.006,
index_1_35.ts?m=1676056675
#EXTINF:6.006,
index_1_36.ts?m=1676056675
#EXTINF:6.006,
index_1_37.ts?m=1676056675
#EXTINF:5.806,
index_1_38.ts?m=1676056675
#EXT-X-DATERANGE:ID="2",START-DATE="2023-02-10T19:20:13.276Z",END-
DATE="2023-02-10T19:21:13.269Z",DURATION=59.993
#EXTINF:6.206,
index_1_39.ts?m=1676056675
#EXTINF:6.006,
index_1_40.ts?m=1676056675
```

Example Lineares personalisiertes HLS-Manifest (mit kreativer Anzeigensignalisierung):

DATERANGE Derjenige, der MediaTailor generiert wird, hat eindeutige ID-Werte. Um die Eindeutigkeit zu gewährleisten (unter Berücksichtigung der in [Mapping SCTE-35 into](#) angegebenen Richtlinien EXT-X-DATERANGE), MediaTailor wird die MEDIA-SEQUENCE Nummer des ersten Anzeigensegments der Avail mit der Sequenznummer der Anzeige innerhalb des Avails verknüpft.

Im folgenden Beispiel verknüpft 421 mit der MediaTailor Positionsnummer der Anzeige. MEDIA-SEQUENCE

```
#EXTM3U
```

```

#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:418
#EXT-X-DISCONTINUITY-SEQUENCE:5
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:55.391Z
#EXTINF:6.006,
https://d3fch9e2fcarly.cloudfront.net/out/v1/1cc7058242a74fdd8aea14e22a9b4131/
index_1_397.ts?m=1676054627
#EXTINF:6.006,
https://d3fch9e2fcarly.cloudfront.net/out/v1/1cc7058242a74fdd8aea14e22a9b4131/
index_1_398.ts?m=1676054627
#EXTINF:5.873,
https://d3fch9e2fcarly.cloudfront.net/out/v1/1cc7058242a74fdd8aea14e22a9b4131/
index_1_399.ts?m=1676054627
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:55.391Z
#EXT-X-DATERANGE:ID="421-1",CLASS="urn:sva:advertising-wg:ad-id-signaling",START-
DATE=2019-01-01T00:02:30.000Z,DURATION=15.015,X-AD-CREATIVE-SIGNALING="base64JSON"
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056813
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056814
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056815
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056816
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056817
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056818
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056819
#EXTINF:1.001,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056820
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:55.391Z

```

```

#EXT-X-DATERANGE:ID="421-1",START-DATE="2023-02-10T19:36:13.435Z",END-
DATE="2023-02-10T19:36:43.432Z",DURATION=15.015
#EXT-X-DATERANGE:ID="421-2",CLASS="urn:sva:advertising-wg:ad-id-signaling",START-
DATE=2019-01-01T00:02:30.000Z,DURATION=15.015,X-AD-CREATIVE-SIGNALING="base64JSON"
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056821
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056822
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056823
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056824
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056825
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056826
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056827
#EXTINF:1.001,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056828
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:55.391Z
#EXT-X-DATERANGE:ID="421-2",START-DATE="2023-02-10T19:36:13.435Z",END-
DATE="2023-02-10T19:36:43.432Z",DURATION=15.015
#EXT-X-DATERANGE:ID="421-3",CLASS="urn:sva:advertising-wg:ad-id-signaling",START-
DATE=2019-01-01T00:02:30.000Z,DURATION=15.015,X-AD-CREATIVE-SIGNALING="base64JSON"
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056829
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056830
#EXTINF:2.002,
../././././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056831
#EXTINF:2.002,

```

```

.././../././segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056832
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056833
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056834
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056835
#EXTINF:1.001,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056836
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-02-10T19:19:55.391Z
#EXT-X-DATERANGE:ID="421-3",START-DATE="2023-02-10T19:36:13.435Z",END-
DATE="2023-02-10T19:36:43.432Z",DURATION=29.997
#EXT-X-DATERANGE:ID="421-4",CLASS="urn:sva:advertising-wg:ad-id-signaling",START-
DATE=2019-01-01T00:02:30.000Z,DURATION=15.015,X-AD-CREATIVE-SIGNALING="base64JSON"
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056837
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056838
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056839
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056840
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056841
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056842
#EXTINF:2.002,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056843
#EXTINF:1.001,
.././.././../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/9e178fa9-
dce5-4248-83d2-5b5d98b019bf/0/1676056844

```



```
#EXT-X-DISCONTINUITY
#EXT-X-DATERANGE:ID="421-4",START-DATE="2023-02-10T19:36:13.435Z",END-
DATE="2023-02-10T19:36:43.432Z",DURATION=15.015
#EXTINF:6.206,
https://d3fch9e2fcarly.cloudfront.net/out/v1/1cc7058242a74fdd8aea14e22a9b4131/
index_1_410.ts?m=1676054627
#EXTINF:6.006,
https://d3fch9e2fcarly.cloudfront.net/out/v1/1cc7058242a74fdd8aea14e22a9b4131/
index_1_411.ts?m=1676054627
```

Example VOD-HLS-Ursprung (mit SCTE-Signalen):

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:1
#EXT-X-PLAYLIST-TYPE:VOD
#EXTINF:6,
index_720p1500k_00001.ts
#EXTINF:6,
index_720p1500k_00002.ts
#EXTINF:6,
index_720p1500k_00003.ts
#EXTINF:6,
index_720p1500k_00004.ts
#EXTINF:6,
index_720p1500k_00005.ts
#EXT-X-CUE-OUT:0
#EXT-X-CUE-IN
#EXTINF:6,
index_720p1500k_00006.ts
#EXTINF:6,
index_720p1500k_00007.ts
#EXTINF:6,
index_720p1500k_00008.ts
#EXTINF:6,
index_720p1500k_00009.ts
#EXTINF:6,
index_720p1500k_00010.ts
#EXTINF:6,
index_720p1500k_00011.ts
#EXTINF:6,
index_720p1500k_00012.ts
```

Example VOD-HLS-Ursprung:

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:1
#EXT-X-PLAYLIST-TYPE:VOD
#EXTINF:6,
index_720p1500k_00001.ts
#EXTINF:6,
index_720p1500k_00002.ts
#EXTINF:6,
index_720p1500k_00003.ts
#EXTINF:6,
index_720p1500k_00004.ts
#EXTINF:4,
index_720p1500k_00005.ts
#EXTINF:2,
index_720p1500k_00006.ts
#EXTINF:6,
index_720p1500k_00007.ts
#EXTINF:6,
index_720p1500k_00008.ts
#EXTINF:6,
index_720p1500k_00009.ts
#EXTINF:6,
index_720p1500k_00010.ts
#EXTINF:6,
index_720p1500k_00011.ts
#EXTINF:6,
index_720p1500k_00012.ts
```

Example Personalisiertes VOD HLS-Manifest:

MediaTailor fügt PROGRAM-DATE-TIME VOD-Manifeste hinzu, um sie als Anker für die DATERANGE HLS-Elemente zu verwenden, die auf Anzeigenpositionen hinweisen.

Das DATERANGE, was MediaTailor generiert wird, hat eindeutige ID-Werte. Um die Eindeutigkeit zu gewährleisten (unter Berücksichtigung der in [Mapping SCTE-35 into](#) angegebenen Richtlinien EXT-X-DATERANGE), MediaTailor wird die MEDIA-SEQUENCE Nummer des ersten Anzeigensegments der Avail mit der Sequenznummer der Anzeige innerhalb des Avails verknüpft.

Im folgenden Beispiel verknüpft 421 mit der MediaTailor Positionsnummer der Anzeige. MEDIA-SEQUENCE

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-PLAYLIST-TYPE:VOD
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:1
#EXT-X-DISCONTINUITY-SEQUENCE:0
#EXT-X-PROGRAM-DATE-TIME:1970-01-01T00:00:00Z
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00001.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00002.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00003.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00004.ts
#EXTINF:4.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00005.ts
#EXT-X-DISCONTINUITY
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/28
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/29
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/30
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/31
#EXTINF:2.002,
../../../../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/32
#EXTINF:2.002,
```

```

../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/33
#EXTINF:2.002,
../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/34
#EXTINF:1.001,
../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/35
#EXT-X-DISCONTINUITY
#EXTINF:2.002,
../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/36
#EXTINF:2.002,
../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/37
#EXTINF:2.002,
../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/38
#EXTINF:2.002,
../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/39
#EXTINF:2.002,
../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/40
#EXTINF:2.002,
../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/41
#EXTINF:2.002,
../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/42
#EXTINF:1.001,
../.../segment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/vod-
variations/9810d863-8736-45fa-866e-be6d2c2bfa20/0/43
#EXT-X-DISCONTINUITY
#EXTINF:2.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00006.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00007.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00008.ts
#EXTINF:6.0,

```

```

https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00009.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00010.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00011.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsco-media/SK0-22/asset-1/hls/
index_720p1500k_00012.ts
#EXT-X-ENDLIST
#EXT-X-DATERANGE:ID="5-1",START-DATE="1970-01-01T00:00:28.000Z",END-
DATE="1970-01-01T00:00:43.015Z",DURATION=15.015
#EXT-X-DATERANGE:ID="5-2",START-DATE="1970-01-01T00:00:43.015Z",END-
DATE="1970-01-01T00:00:58.030Z",DURATION=15.01

```

Personalisierung von DASH-Manifesten mit Anzeigenmetadaten

MediaTailor personalisiert das Manifest mit den vom Ad Decision Server (ADS) zurückgegebenen Creatives. Enthält für jede Anzeige MediaTailor auch ein EventStream Element, das sich über die gesamte Laufzeit der Anzeige erstreckt. Das Event Elementformat ähnelt dem Format, das im Abschnitt [Ad Creative Signaling in DASH und HLS](#) in der Version 2023 der technischen Publikation von SVA beschrieben wird.

Bei unzureichend ausgefüllten Werbeunterbrechungen in Konfigurationen, bei denen Slate aktiviert ist, wird der Slate-Zeitraum an das Ende des Verfügbarkeitszeitraums MediaTailor angehängt, jedoch ohne Metadaten EventStream

Fügt für jede Anzeige, die in das personalisierte Manifest eingebunden wird, die kreativen Metadaten MediaTailor hinzu, die als Element innerhalb eines CDATA Elements dargestellt werden. Event

Example Linearer DASH-Ursprung (Inline-SCTE-Attribute):

```

<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd" id="201"
  type="dynamic" publishTime="2023-02-10T21:08:40+00:00" minimumUpdatePeriod="PT6S"
  availabilityStartTime="2023-02-09T22:47:05.865000+00:00" minBufferTime="PT10S"
  suggestedPresentationDelay="PT20.000S" timeShiftBufferDepth="PT88.999S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011">

```

```

<Period start="PT80141.456S" id="104" duration="PT304.103S">
  <AdaptationSet id="1485523442" mimeType="video/mp4" segmentAlignment="true"
startWithSAP="1" subsegmentAlignment="true" subsegmentStartsWithSAP="1"
bitstreamSwitching="true">
    <SegmentTemplate timescale="60000" media="index_video_$RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_video_$RepresentationID$_0_init.mp4?
m=1676062374" startNumber="151" presentationTimeOffset="4808487386">
        <SegmentTimeline>
            <S t="4824975858" d="360360" r="3"/>
            <S t="4826417298" d="316316"/>
        </SegmentTimeline>
    </SegmentTemplate>
    <Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="1800000" codecs="avc1.4D401F"/>
    <Representation id="3" width="640" height="360" frameRate="30000/1001"
bandwidth="1200000" codecs="avc1.4D401E"/>
    <Representation id="5" width="480" height="270" frameRate="30000/1001"
bandwidth="800000" codecs="avc1.4D4015"/>
  </AdaptationSet>
  <AdaptationSet id="1377232898" mimeType="audio/mp4" segmentAlignment="0"
lang="eng">
    <Label>eng</Label>
    <SegmentTemplate timescale="48000" media="index_audio_$RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_audio_$RepresentationID$_0_init.mp4?
m=1676062374" startNumber="151" presentationTimeOffset="3846790126">
        <SegmentTimeline>
            <S t="3859981294" d="287744"/>
            <S t="3860269038" d="288768"/>
            <S t="3860557806" d="287744"/>
            <S t="3860845550" d="288768"/>
            <S t="3861134318" d="252928"/>
        </SegmentTimeline>
    </SegmentTemplate>
    <Representation id="2" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
    <Representation id="4" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
  </AdaptationSet>
</Period>

```

```

    <Representation id="6" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
      <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
  </AdaptationSet>
  <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2023-02-10T21:02:31.007Z"/>
</Period>
<Period start="PT80445.560S" id="155" duration="PT44.978S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="4048044">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="207000"
tier="4095">
        <scte35:SpliceInsert spliceEventId="111" spliceEventCancelIndicator="false"
outOfNetworkIndicator="true" spliceImmediateFlag="false" uniqueProgramId="1"
availNum="1" availsExpected="1">
          <scte35:Program>
            <scte35:SpliceTime ptsTime="7239893422"/>
          </scte35:Program>
          <scte35:BreakDuration autoReturn="true" duration="4048044"/>
        </scte35:SpliceInsert>
      </scte35:SpliceInfoSection>
    </Event>
  </EventStream>
  <AdaptationSet id="1485523442" mimeType="video/mp4" segmentAlignment="true"
startWithSAP="1" subsegmentAlignment="true" subsegmentStartsWithSAP="1"
bitstreamSwitching="true">
    <SegmentTemplate timescale="60000" media="index_video_${RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_video_${RepresentationID$_0_init}.mp4?
m=1676062374" startNumber="156" presentationTimeOffset="4826733614">
      <SegmentTimeline>
        <S t="4826733614" d="284284"/>
        <S t="4827017898" d="360360" r="5"/>
        <S t="4829180058" d="252252"/>
      </SegmentTimeline>
    </SegmentTemplate>
    <Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="1800000" codecs="avc1.4D401F"/>
    <Representation id="3" width="640" height="360" frameRate="30000/1001"
bandwidth="1200000" codecs="avc1.4D401E"/>
    <Representation id="5" width="480" height="270" frameRate="30000/1001"
bandwidth="800000" codecs="avc1.4D4015"/>
  </AdaptationSet>

```

```

    <AdaptationSet id="1377232898" mimeType="audio/mp4" segmentAlignment="0"
    lang="eng">
      <Label>eng</Label>
      <SegmentTemplate timescale="48000" media="index_audio_$RepresentationID$_
$Number$.mp4?m=1676062374" initialization="index_audio_$RepresentationID$_init.mp4?
m=1676062374" startNumber="156" presentationTimeOffset="3861387246">
        <SegmentTimeline>
          <S t="3861387246" d="227328"/>
          <S t="3861614574" d="288768"/>
          <S t="3861903342" d="287744"/>
          <S t="3862191086" d="288768"/>
          <S t="3862479854" d="287744"/>
          <S t="3862767598" d="288768"/>
          <S t="3863056366" d="287744"/>
          <S t="3863344110" d="202752"/>
        </SegmentTimeline>
      </SegmentTemplate>
      <Representation id="2" bandwidth="193007" audioSamplingRate="48000"
      codecs="mp4a.40.2">
        <AudioChannelConfiguration
        schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
      </Representation>
      <Representation id="4" bandwidth="193007" audioSamplingRate="48000"
      codecs="mp4a.40.2">
        <AudioChannelConfiguration
        schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
      </Representation>
      <Representation id="6" bandwidth="193007" audioSamplingRate="48000"
      codecs="mp4a.40.2">
        <AudioChannelConfiguration
        schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
      </Representation>
    </AdaptationSet>
    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
    value="2023-02-10T21:07:35.111Z"/>
  </Period>
  <Period start="PT80490.538S" id="163">
    <AdaptationSet id="1485523442" mimeType="video/mp4" segmentAlignment="true"
    startWithSAP="1" subsegmentAlignment="true" subsegmentStartsWithSAP="1"
    bitstreamSwitching="true">
      <SegmentTemplate timescale="60000" media="index_video_$RepresentationID$_
$Number$.mp4?m=1676062374" initialization="index_video_$RepresentationID$_init.mp4?
m=1676062374" startNumber="164" presentationTimeOffset="4829432310">
        <SegmentTimeline>

```



```

        <S t="4829432310" d="348348"/>
        <S t="4829780658" d="360360" r="1"/>
    </SegmentTimeline>
</SegmentTemplate>
    <Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="1800000" codecs="avc1.4D401F"/>
    <Representation id="3" width="640" height="360" frameRate="30000/1001"
bandwidth="1200000" codecs="avc1.4D401E"/>
    <Representation id="5" width="480" height="270" frameRate="30000/1001"
bandwidth="800000" codecs="avc1.4D4015"/>
</AdaptationSet>
    <AdaptationSet id="1377232898" mimeType="audio/mp4" segmentAlignment="0"
lang="eng">
        <Label>eng</Label>
        <SegmentTemplate timescale="48000" media="index_audio_$RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_audio_$RepresentationID$_0_init.mp4?
m=1676062374" startNumber="164" presentationTimeOffset="3863546862">
            <SegmentTimeline>
                <S t="3863546862" d="278528"/>
                <S t="3863825390" d="287744"/>
                <S t="3864113134" d="288768"/>
            </SegmentTimeline>
        </SegmentTemplate>
        <Representation id="2" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
        <Representation id="4" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
        <Representation id="6" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
    </AdaptationSet>
    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2023-02-10T21:08:20.090Z"/>
</Period>
</MPD>

```

Example Personalisiertes lineares DASH-Manifest (mit kreativer Anzeigensignalisierung):

```

<MPD availabilityStartTime="2023-02-09T22:47:05.865000+00:00"
  id="201" minBufferTime="PT10S" minimumUpdatePeriod="PT6S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011"
  publishTime="2023-02-10T21:08:43+00:00" suggestedPresentationDelay="PT20.000S"
  timeShiftBufferDepth="PT88.999S" type="dynamic" xmlns="urn:mpeg:dash:schema:mpd:2011"
  xmlns:scte35="urn:scte:scte35:2013:xml" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
  instance" xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd">
  <BaseURL>https://d3fch9e2fcarly.cloudfront.net/out/v1/
  f9f38deca3f14fc4b5ab3cdbd76cfb9e/</BaseURL>
  <Location>https://777788889999.mediatailor.us-west-2.amazonaws.com/
  v1/dash/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/out/v1/
  f9f38deca3f14fc4b5ab3cdbd76cfb9e/index.mpd?
  aws.sessionId=672ed481-4ffd-4270-936f-7c8403947f2e</Location>
  <Period duration="PT304.103S" id="104" start="PT80141.456S">
    <AdaptationSet bitstreamSwitching="true" id="1485523442" mimeType="video/
    mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
    subsegmentStartsWithSAP="1">
      <SegmentTemplate initialization="index_video_$RepresentationID$_init.mp4?
      m=1676062374" media="index_video_$RepresentationID$_$Number$.mp4?m=1676062374"
      presentationTimeOffset="4808487386" startNumber="151" timescale="60000">
        <SegmentTimeline>
          <S d="360360" r="3" t="4824975858"/>
          <S d="316316" t="4826417298"/>
        </SegmentTimeline>
      </SegmentTemplate>
      <Representation bandwidth="1800000" codecs="avc1.4D401F"
      frameRate="30000/1001" height="540" id="1" width="960"/>
      <Representation bandwidth="1200000" codecs="avc1.4D401E"
      frameRate="30000/1001" height="360" id="3" width="640"/>
      <Representation bandwidth="800000" codecs="avc1.4D4015"
      frameRate="30000/1001" height="270" id="5" width="480"/>
    </AdaptationSet>
    <AdaptationSet id="1377232898" lang="eng" mimeType="audio/mp4"
    segmentAlignment="0">
      <Label>eng</Label>
      <SegmentTemplate initialization="index_audio_$RepresentationID$_init.mp4?
      m=1676062374" media="index_audio_$RepresentationID$_$Number$.mp4?m=1676062374"
      presentationTimeOffset="3846790126" startNumber="151" timescale="48000">
        <SegmentTimeline>
          <S d="287744" t="3859981294"/>
          <S d="288768" t="3860269038"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </AdaptationSet>
  </Period>
</MPD>

```

```

        <S d="287744" t="3860557806"/>
        <S d="288768" t="3860845550"/>
        <S d="252928" t="3861134318"/>
    </SegmentTimeline>
</SegmentTemplate>
<Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="2">
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
</Representation>
<Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="4">
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
</Representation>
<Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="6">
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
</Representation>
</AdaptationSet>
<SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2023-02-10T21:02:31.007Z"/>
</Period>
<Period id="155_1" start="PT22H20M45.56S">
    <BaseURL>https://777788889999.mediatailor.us-west-2.amazonaws.com/
v1/dashsegment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/
emt/672ed481-4ffd-4270-936f-7c8403947f2e/155/155_1/</BaseURL>
    <EventStream schemeIdUri="urn:sva:advertising-wg:ad-id-signaling"
timescale="90000">
        <Event presentationTime="xxxxx" duration="1351350">
            <![CDATA[{"version": 1,"identifiers": [{"scheme":
"urn:smp:ul:060E2B34.01040101.01200900.00000000","value": "155_1","ad_position":
"155_1", "ad_type":"avail","creative_id": "123","tracking_uri": "../v1/
tracking/hashed-account-id/origin-id/session-id","custom_vast_data":"123abc"}]]]>
        </Event>
    </EventStream>
    <AdaptationSet bitstreamSwitching="false" frameRate="30000/1001"
mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SegmentTemplate startNumber="1" timescale="90000"/>
        <Representation bandwidth="1800000" codecs="avc1.64001f" height="540"
id="1" width="960">

```

```

        <SegmentTemplate initialization="asset_540_1_2init.mp4"
media="asset_540_1_2_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
            <S d="180180" r="6" t="0"/>
            <S d="90090" t="1261260"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="1200000" codecs="avc1.64001e" height="360"
id="3" width="640">
    <SegmentTemplate initialization="asset_360_1_1init.mp4"
media="asset_360_1_1_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
            <S d="180180" r="6" t="0"/>
            <S d="90090" t="1261260"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation bandwidth="800000" codecs="avc1.640015" height="270" id="5"
width="480">
    <SegmentTemplate initialization="asset_270_0_0init.mp4"
media="asset_270_0_0_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
            <S d="180180" r="6" t="0"/>
            <S d="90090" t="1261260"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000"/>
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="6">
        <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000">
            <SegmentTimeline>
                <S d="98304" t="0"/>
                <S d="96256" r="1" t="98304"/>
                <S d="95232" t="290816"/>
                <S d="96256" r="2" t="386048"/>
                <S d="48128" t="674816"/>
            </SegmentTimeline>

```

```

        </SegmentTemplate>
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
    </AdaptationSet>
</Period>
<Period id="155_2" start="PT22H21M0.575S">
    <BaseURL>https://777788889999.mediatailor.us-west-2.amazonaws.com/
v1/dashsegment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/
emt/672ed481-4ffd-4270-936f-7c8403947f2e/155/155_2/</BaseURL>
    <EventStream schemeIdUri="urn:sva:advertising-wg:ad-id-signaling"
timescale="90000">
        <Event presentationTime="0" duration="1351350">
            <![CDATA[{"version": 1,"identifiers": [{"scheme":
"urn:smp:ul:060E2B34.01040101.01200900.00000000","value": "155_2","ad_position":
"155_2", "ad_type":"avail","creative_id": "234","tracking_uri": "../.../v1/
tracking/hashed-account-id/origin-id/session-id","custom_vast_data":"123abc"}]]]>
        </Event>
    </EventStream>
    <AdaptationSet bitstreamSwitching="false" frameRate="30000/1001"
mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SegmentTemplate startNumber="1" timescale="90000"/>
        <Representation bandwidth="1800000" codecs="avc1.64001f" height="540"
id="1" width="960">
            <SegmentTemplate initialization="asset_540_1_2init.mp4"
media="asset_540_1_2_${Number%09d$.mp4" startNumber="1" timescale="90000">
                <SegmentTimeline>
                    <S d="180180" r="6" t="0"/>
                    <S d="90090" t="1261260"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="1200000" codecs="avc1.64001e" height="360"
id="3" width="640">
            <SegmentTemplate initialization="asset_360_1_1init.mp4"
media="asset_360_1_1_${Number%09d$.mp4" startNumber="1" timescale="90000">
                <SegmentTimeline>
                    <S d="180180" r="6" t="0"/>
                    <S d="90090" t="1261260"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </Period>
</Manifest>

```

```
<Representation bandwidth="800000" codecs="avc1.640015" height="270" id="5"
width="480">
  <SegmentTemplate initialization="asset_270_0_0init.mp4"
media="asset_270_0_0_${Number%09d$.mp4" startNumber="1" timescale="90000">
    <SegmentTimeline>
      <S d="180180" r="6" t="0"/>
      <S d="90090" t="1261260"/>
    </SegmentTimeline>
  </SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
  <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000"/>
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="6">
      <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000">
        <SegmentTimeline>
          <S d="98304" t="0"/>
          <S d="96256" r="1" t="98304"/>
          <S d="95232" t="290816"/>
          <S d="96256" r="2" t="386048"/>
          <S d="48128" t="674816"/>
        </SegmentTimeline>
      </SegmentTemplate>
      <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
  </AdaptationSet>
</Period>
<Period id="155_3" start="PT22H21M15.59S">
  <BaseURL>https://7777888899999.mediataylor.us-west-2.amazonaws.com/
v1/dashsegment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/
emt/672ed481-4ffd-4270-936f-7c8403947f2e/155/155_3/</BaseURL>
  <EventStream schemeIdUri="urn:sva:advertising-wg:ad-id-signaling"
timescale="90000">
    <Event presentationTime="0" duration="1351350">
      <![CDATA[{"version": 1,"identifiers": [{"scheme":
"urn:smp:ul:060E2B34.01040101.01200900.00000000","value": "155_3","ad_position":
"155_3", "ad_type":"avail","creative_id": "345","tracking_uri": "../../v1/
tracking/hashed-account-id/origin-id/session-id","custom_vast_data":"123abc"}]]]>
    </Event>
  </EventStream>
</Period>
</AdaptationSet>
</Representation>
</Media>
</MPD>
```

```

    </EventStream>
    <AdaptationSet bitstreamSwitching="false" frameRate="30000/1001"
    mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
    subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SegmentTemplate startNumber="1" timescale="90000"/>
        <Representation bandwidth="1800000" codecs="avc1.64001f" height="540"
    id="1" width="960">
            <SegmentTemplate initialization="asset_540_1_2init.mp4"
    media="asset_540_1_2_${Number%09d$.mp4" startNumber="1" timescale="90000">
                <SegmentTimeline>
                    <S d="180180" r="6" t="0"/>
                    <S d="90090" t="1261260"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="1200000" codecs="avc1.64001e" height="360"
    id="3" width="640">
            <SegmentTemplate initialization="asset_360_1_1init.mp4"
    media="asset_360_1_1_${Number%09d$.mp4" startNumber="1" timescale="90000">
                <SegmentTimeline>
                    <S d="180180" r="6" t="0"/>
                    <S d="90090" t="1261260"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="800000" codecs="avc1.640015" height="270" id="5"
    width="480">
            <SegmentTemplate initialization="asset_270_0_0init.mp4"
    media="asset_270_0_0_${Number%09d$.mp4" startNumber="1" timescale="90000">
                <SegmentTimeline>
                    <S d="180180" r="6" t="0"/>
                    <S d="90090" t="1261260"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>
    <AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
        <SegmentTemplate initialization="asset_audio_128_3init.mp4"
    media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000"/>
        <Label>eng</Label>
        <Representation audioSamplingRate="48000" bandwidth="128000"
    codecs="mp4a.40.2" id="6">
            <SegmentTemplate initialization="asset_audio_128_3init.mp4"
    media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000">

```

```

        <SegmentTimeline>
            <S d="98304" t="0"/>
            <S d="96256" r="1" t="98304"/>
            <S d="95232" t="290816"/>
            <S d="96256" r="2" t="386048"/>
            <S d="48128" t="674816"/>
        </SegmentTimeline>
    </SegmentTemplate>
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
</AdaptationSet>
</Period>
<Period id="163" start="PT80490.538S">
    <AdaptationSet bitstreamSwitching="true" id="1485523442" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
        <SegmentTemplate initialization="index_video_$RepresentationID$_init.mp4?
m=1676062374" media="index_video_$RepresentationID$_$Number$.mp4?m=1676062374"
presentationTimeOffset="4829432310" startNumber="164" timescale="60000">
            <SegmentTimeline>
                <S d="348348" t="4829432310"/>
                <S d="360360" r="1" t="4829780658"/>
            </SegmentTimeline>
        </SegmentTemplate>
        <Representation bandwidth="1800000" codecs="avc1.4D401F"
frameRate="30000/1001" height="540" id="1" width="960"/>
        <Representation bandwidth="1200000" codecs="avc1.4D401E"
frameRate="30000/1001" height="360" id="3" width="640"/>
        <Representation bandwidth="800000" codecs="avc1.4D4015"
frameRate="30000/1001" height="270" id="5" width="480"/>
    </AdaptationSet>
    <AdaptationSet id="1377232898" lang="eng" mimeType="audio/mp4"
segmentAlignment="0">
        <Label>eng</Label>
        <SegmentTemplate initialization="index_audio_$RepresentationID$_init.mp4?
m=1676062374" media="index_audio_$RepresentationID$_$Number$.mp4?m=1676062374"
presentationTimeOffset="3863546862" startNumber="164" timescale="48000">
            <SegmentTimeline>
                <S d="278528" t="3863546862"/>
                <S d="287744" t="3863825390"/>
                <S d="288768" t="3864113134"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </AdaptationSet>
</Period>
</MPD>

```



```

        <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="2">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
        <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="4">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
        <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="6">
            <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
        </Representation>
    </AdaptationSet>
    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2023-02-10T21:08:20.090Z"/>
</Period>
</MPD>

```

Interaktionen mit dem Ad Decision Server (ADS)

MediaTailor verwendet den Wert des kreativen id Attributs aus der VAST-Antwort als Wert für die Signalisierung der Anzeigen-ID. Wenn der id Attributwert leer oder in der VAST-Antwort nicht vorhanden ist, MediaTailor wird ein leerer Wert in die Signalisierung der Anzeigen-ID eingefügt.

Example VAST-Antwort:

Die folgende VAST-Beispielantwort enthält einen id Attributwert für die lineare Inline-AntwortCreative. MediaTailor extrahiert den Wert aus dem benutzerdefinierten Extension VAST-Element und fügt diesen Wert in die kreativen Metadaten des Manifests ein.

```

<?xml version="1.0" encoding="utf-8"?>
<VAST version="3.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <Ad sequence="3">
        <Inline>
            <AdSystem>2.0</AdSystem>
            <AdTitle>AD-caribbean2-15</AdTitle>
            <Impression><![CDATA[https://n8ljfs0xxx.execute-api.us-
west-2.amazonaws.com/v1/impression]]></Impression>
            <Creatives>

```

```

    <Creative sequence="3" apiFramework="inLine" id="1234">
      <Linear>
        <Duration>00:00:15</Duration>
        <MediaFiles>
          <MediaFile id="00002" delivery="progressive" type="video/
mp4" width="1280" height="720"><![CDATA[https://d3re4i3vgppxxx.cloudfront.net/Media/
Bumpers/AD-caribbean2-15-HD.mp4]]></MediaFile>
        </MediaFiles>
      </Linear>
    </Creative>
  </Creatives>
  <Extensions>
    <Extension type="creative_signaling"><![CDATA[999999|
TVN1DDNpFTchtpRj,E5TfTtcYd5IEzvEt,ChA050HcvWRGFY6Zp5VSS1xUEJ2B9p8GGhQIDzIQkFeQC-
Ho67FR3P9qNa6khSAGKgAyAA]]></Extension>
  </Extensions>
</InLine>
</Ad>
</VAST>

```

Client-seitige Tracking-API

Das folgende Beispiel zeigt, wie ein Player-SDK die Anzeigenmetadaten im Manifest mit den vollständigen Tracking-Eventdaten in der clientseitigen Tracking-Antwort-Payload mit und verknüpft. `creativeId` `adId`

Example JSON-Nachricht:

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "ads": [
        {
          "adId": "5",
          "adParameters": "",
          "adProgramDateTime": null,
          "adSystem": "2.0",
          "adTitle": "AD-caribbean2-15",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "1234",
          "creativeSequence": "2",

```

```

    "duration": "PT15S",
    "durationInSeconds": 15,
    "extensions": [],
    "mediaFiles": {
      "mediaFilesList": [],
      "mezzanine": ""
    },
    "skipOffset": null,
    "startTime": "PT30S",
    "startTimeInSeconds": 30,
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://myServer/impression"
        ],
        "duration": "PT15S",
        "durationInSeconds": 15,
        "eventId": "5",
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT30S",
        "startTimeInSeconds": 30
      }
    ],
    "vastAdId": ""
  }
],
"availId": "5",
"availProgramDateTime": null,
"duration": "PT15S",
"durationInSeconds": 15,
"meta": null,
"nonLinearAdsList": [],
"startTime": "PT30S",
"startTimeInSeconds": 30
}
],
"nextToken": "UFQ1TTM0Ljk2N1NfMjAyMi0xMS0xOFQwNDozMzo1Mi4yNDUxOTdaXzE%3D",
"nonLinearAvails": []
}

```

Wird verwendet AWS Elemental MediaTailor , um lineare zusammengestellte Streams zu erstellen

AWS Elemental MediaTailor Channel Assembly ist ein reiner Dienst, mit dem Sie lineare Streaming-Kanäle erstellen können, indem Sie Ihre vorhandenen Video-on-Demand-Inhalte (VOD) mit Live-Inhalten mischen. MediaTailor berührt niemals Ihre Inhaltssegmente, die direkt von Ihrem Ursprungsserver bereitgestellt werden. MediaTailor Ruft stattdessen die Manifeste von Ihrem Ursprung ab und verwendet sie, um ein Live-Slide-Manifestfenster zusammenzustellen, das auf die zugrunde liegenden Inhaltssegmente verweist. Bei der Kanalzusammenstellung werden Dinge wie die Mediensequenznummer gespeichert, die für eine reibungslose Wiedergabe von Asset zu Asset erforderlich ist. Linear zusammengestellte Streams werden mit geringen Betriebskosten erstellt, indem vorhandene, mit mehreren Bitraten kodierte und verpackte VOD-Inhalte verwendet werden.

Sie können lineare Channel-Assembly-Streams ganz einfach monetarisieren, indem Sie Werbeunterbrechungen in Ihre Programme einfügen, ohne den Inhalt mit SCTE-35-Markern konditionieren zu müssen. Sie können Channel Assembly zusammen mit dem Dienst zur MediaTailor Anzeigeneinfügung oder einem beliebigen serverseitigen Dienst zum Einfügen von Anzeigen verwenden.

Informationen zu den ersten Schritten mit der Kanalmontage finden Sie unter [the section called “Erste Schritte mit der MediaTailor Kanalmontage”](#).

Themen

- [Mit Quellstandorten arbeiten](#)
- [Mit Kanälen arbeiten](#)
- [Ein Programm zum Zeitplan eines Kanals hinzufügen](#)
- [Fügen Sie personalisierte Anzeigen und Werbeunterbrechungen in einen Channel-Stream ein](#)
- [Zeitversetzte Anzeige aktivieren](#)
- [Behebung von Wiedergabefehlern, die zurückgegeben wurden von MediaTailor](#)

Mit Quellstandorten arbeiten

Ein Quellspeicherort stellt den Ursprungsserver dar, auf dem Ihr Quellinhalt gespeichert ist. Ein Quellstandort kann Amazon S3, ein Standard-Webserver, ein Content Delivery Network (CDN)

wie Amazon CloudFront oder ein Verpackungshersteller wie AWS Elemental MediaPackage sein. MediaTailor ruft Ihre Inhaltsmanifeste vom Quellspeicherort ab und verwendet sie, um den linearen Stream Ihres Kanals zusammenzustellen.

In diesem Thema wird erklärt, wie Sie mit der AWS Elemental MediaTailor Konsole Quellverzeichnisse erstellen und löschen und wie Sie mit VOD-Quellen arbeiten.

Themen

- Einem Quellspeicherort erstellen
- Konfiguration der Authentifizierung für Ihren Quellstandort
- Mit VOD-Quellen arbeiten
- Arbeiten mit Live-Quellen
- Verwenden von Paketkonfigurationen
- Zwischenspeichern von Manifesten

Einen Quellspeicherort erstellen

Im folgenden Verfahren wird erklärt, wie Sie mithilfe der MediaTailor Konsole einen Quellspeicherort erstellen. Informationen zum Erstellen von Quellspeicherorten mithilfe der MediaTailor API finden Sie [CreateSourceLocation](#) in der AWS Elemental MediaTailor API-Referenz.

So erstellen Sie einen Quellspeicherort

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Source Locations aus.
3. Wählen Sie in der Navigationsleiste die Option Quellspeicherort erstellen aus.
4. Geben Sie unter Konfiguration des Quellverzeichnis einen Namen und die Basis-URL Ihres Ursprungsservers ein:
 - Name: Eine Kennung für Ihren Quellstandort, z. B. my-origin.
 - Basis-URL: Das Protokoll und die Basis-URL des Ursprungsservers, auf dem Ihre Inhalte gespeichert sind, z. B. <https://111111111111.cloudfront.net>. Die URL muss in einem Standard-HTTP-URL-Format mit dem Präfix <http://> oder <https://> vorliegen.

Wählen Sie optional SigV4 für die Amazon S3 S3-Authentifizierung verwenden, wenn Ihr Quellstandort ein Amazon S3 S3-Bucket ist und Sie AWS Signature Version 4 für die Amazon

S3 S3-Zugriffsauthentifizierung verwenden möchten. Weiterführende Informationen finden Sie unter [Konfiguration der Authentifizierung für Ihren Quellstandort](#).

5.

Konfigurieren Sie unter Zugriffskonfiguration optional die Authentifizierung für Ihren Quellstandort:

- Zugriffstyp: Wählen Sie den Authentifizierungstyp aus, der für den Zugriff auf den Inhalt MediaTailor verwendet wird, der am Quellspeicherort gespeichert ist.
- SigV4 für Amazon S3 — MediaTailor verwendet Amazon Signature Version 4 (SigV4), um Anfragen an Ihren Absender zu autorisieren. Weitere Informationen finden Sie unter [the section called “Authentifizieren von Anfragen an Amazon S3 mit SigV4”](#).
- Secrets Manager-Zugriffstoken-Authentifizierung — MediaTailor verwendet Secrets Manager und einen vom AWS KMS Kunden verwalteten Schlüssel, der von Ihnen erstellt wurde, in Ihrem Besitz ist und verwaltet wird, um die Zugriffstoken-Authentifizierung zwischen MediaTailor und Ihrem Ursprung zu erleichtern. Informationen zur Konfiguration der Secrets Manager Manager-Zugriffstoken-Authentifizierung finden Sie unter [the section called “AWS Secrets Manager Arbeitet mit Zugriffstoken-Authentifizierung”](#).
- Header-Name — Geben Sie einen HTTP-Header-Namen an. MediaTailor verwendet den HTTP-Header, um das Zugriffstoken in Inhaltsmanifestanfragen an Ihren Ursprung zu senden. Sie können jeden Header-Namen verwenden, solange er nicht mit x-amz- oder beginnntx-amzn-. Wenn Sie eine Integration mit [MediaPackage CDN-Autorisierung durchführen](#), sollte der Header-Wert wie folgt lautenX-MediaPackage-CDNIdentifizier.
- Geheimer Zeichenkettenschlüssel — Der SecretString Schlüssel, den Sie in Ihrem Secrets Manager Manager-Geheimnis angegeben haben. Wenn Ihr beispielsweise ein Schlüssel- und Wertepaar wie: SecretString enthält{"MyHeaderName": "11111111-2222-3333-4444-111122223333"}, dann MyHeaderName ist es der SecretString Schlüssel, den Sie in dieses Feld eingeben.
- Geheimer ARN — Der ARN des Geheimnisses, das Ihr Zugriffstoken enthält. Eine step-by-step Anleitung finden Sie unter [Schritt 2: Erstellen Sie ein AWS Secrets Manager Geheimnis](#).

6. Konfigurieren Sie unter Konfiguration des Segmentbereitstellungsservers optional einen Server für die Bereitstellung Ihrer Inhaltssegmente:

- Verwenden Sie einen Standardserver für die Segmentzustellung: Geben Sie die Basis-URL des Servers ein, der für die Bereitstellung Ihrer Inhaltssegmente verwendet wird, z. B. ein CDN. Konfigurieren Sie den Standard-Hostnamen für das Segment, wenn Sie für die

Bereitstellung der Inhaltssegmente einen anderen Server als den Quellserver verwenden möchten. Sie können beispielsweise den Zugriff von Spielern auf die Ursprungsmanifeste einschränken, indem Sie eine andere CDN-Konfiguration für die Basis-HTTP-URL (die MediaTailor verwendet, um auf die Manifeste zuzugreifen) und die Standardsegment-Basis-URL (die der Spieler für den Zugriff auf die Inhaltssegmente verwendet) verwenden. Wenn Sie keinen Wert eingeben, wird MediaTailor standardmäßig der Quellserver für die Segmentzustellung verwendet.

- Verwenden Sie benannte Segmentzustellungsserver: Wenn Sie einen Standardserver für die Segmentzustellung konfiguriert haben, können Sie auch zusätzliche Server für die Segmentzustellung konfigurieren. Jeder muss einen eindeutigen Namen und eine Basis-URL haben. Die Basis-URL kann eine vollständige HTTP-URL sein oder ein relativer Pfad wie `/some/path/`. Die Namen werden verwendet, um zu identifizieren, welcher Server verwendet werden soll, wenn MediaTailor er eine Anfrage für Inhaltssegmente erhält. Wenn die Anfrage den Header enthält `X-MediaTailor-SegmentDeliveryConfigurationName` und der Wert des Headers mit einem Namen übereinstimmt, wird die entsprechende Basis-URL verwendet, um den Inhalt bereitzustellen. Wenn der Header nicht in der Anfrage enthalten ist oder wenn er keinem Namen entspricht, wird der Standardserver für die Segmentzustellung verwendet.

7. Wählen Sie Quellspeicherort erstellen aus.

8. Um weitere Quellstandorte hinzuzufügen, wiederholen Sie die Schritte 2-6.

Konfiguration der Authentifizierung für Ihren Quellstandort

Verwenden Sie die Zugriffskonfiguration, um die Authentifizierung für Ihren Quellstandort zu konfigurieren. Wenn die Zugriffskonfiguration aktiviert ist, werden Quellmanifeste von Ihrem Ursprung MediaTailor nur abgerufen, wenn die Anfrage zwischen MediaTailor und Ihrem Ursprung autorisiert wurde. Die Zugriffskonfiguration ist standardmäßig ausgeschaltet.

MediaTailor unterstützt die folgenden Authentifizierungstypen:

- SigV4 für Amazon S3 S3-Authentifizierung
- AWS Secrets Manager Zugriffstoken
- SigV4 für die MediaPackage Authentifizierung der Version 2 (v2)

In diesem Kapitel wird erklärt, wie SigV4 für Amazon S3, MediaPackage v2 und AWS Secrets Manager Zugriffstoken für die Quellstandortauthentifizierung verwendet werden.

Für weitere Informationen wählen Sie das entsprechende Thema aus.

Themen

- [Authentifizieren von Anfragen an Amazon S3 mit SigV4](#)
- [Arbeiten mit SigV4 für Version 2 MediaPackage](#)
- [AWS Secrets Manager Arbeitet mit Zugriffstoken-Authentifizierung](#)

Authentifizieren von Anfragen an Amazon S3 mit SigV4

Signature Version 4 (Sigv4) für Amazon S3 ist ein Signaturprotokoll, das zur Authentifizierung von Anfragen an Amazon S3 über HTTPS verwendet wird. Wenn Sie SigV4 für Amazon S3 verwenden, MediaTailor fügt es einen signierten Autorisierungsheader in die HTTPS-Anfrage an den Amazon S3 S3-Bucket ein, der als Ihr Ursprung verwendet wird. Wenn der signierte Autorisierungsheader gültig ist, erfüllt Ihr Absender die Anfrage. Wenn er nicht gültig ist, schlägt die Anfrage fehl.

Allgemeine Informationen zu SigV4 für AWS Key Management Service finden Sie im Thema [Authenticating Requests \(AWS Signature Version 4\)](#) in der Amazon S3 S3-API-Referenz.

Note

MediaTailor signiert Anfragen an diese Ursprünge immer mit SigV4.

Voraussetzungen

Wenn Sie die SigV4-Authentifizierung für Amazon S3 für Ihren Quellstandort aktivieren, müssen Sie die folgenden Anforderungen erfüllen:

- Sie müssen MediaTailor den Zugriff auf Ihren Amazon S3 S3-Bucket zulassen, indem Sie [mediatailor.amazonaws.com](#) den Hauptzugriff in IAM gewähren. [Informationen zur Konfiguration des Zugriffs in IAM finden Sie unter Zugriffsverwaltung im Benutzerhandbuch.AWS Identity and Access Management](#)
- Der Service Principal von [mediatailor.amazonaws.com](#) muss berechtigt sein, alle Manifeste der obersten Ebene zu lesen, auf die in den VOD-Quellpaketkonfigurationen verwiesen wird.
- Der Aufrufer der API muss über s3: GetObject IAM-Berechtigungen verfügen, um alle Manifeste der obersten Ebene lesen zu können, auf die in Ihren VOD-Quellpaketkonfigurationen verwiesen wird. MediaTailor

- Ihre Basis-URL für den MediaTailor Quellstandort muss dem Anforderungs-URL-Format von Amazon S3 im virtuellen Hosted-Stil entsprechen. Zum Beispiel `https://bucket-name.s3.Region.amazonaws.com/.key-name` Informationen zum von Amazon S3 gehosteten virtuellen Zugriff finden Sie unter Anfragen im [virtuellen Hosted-Stil](#).

Arbeiten mit SigV4 für Version 2 MediaPackage

Signature Version 4 (Sigv4) für MediaPackage v2 ist ein Signaturprotokoll, das zur Authentifizierung von Anfragen an MediaPackage v2 über HTTP verwendet wird. Wenn Sie SigV4 für MediaPackage v2 verwenden, MediaTailor fügt es einen signierten Autorisierungsheader in die HTTP-Anfrage an den MediaPackage v2-Endpunkt ein, der als Ihr Ursprung verwendet wird. Wenn der signierte Autorisierungsheader gültig ist, erfüllt Ihr Absender die Anfrage. Wenn er nicht gültig ist, schlägt die Anfrage fehl.

Allgemeine Informationen zu SigV4 für MediaPackage v2 finden Sie im Thema [Authenticating Requests \(AWS Signature Version 4\)](#) in der MediaPackage v2-API-Referenz.

Voraussetzungen

Wenn Sie die SigV4-Authentifizierung für MediaPackage Version 2 für Ihren Quellstandort aktivieren, müssen Sie die folgenden Anforderungen erfüllen:

- Sie müssen MediaTailor den Zugriff auf Ihren MediaPackage v2-Endpunkt zulassen, indem Sie `mediatailor.amazonaws.com` in einer Origin-Zugriffsrichtlinie für den Endpunkt den Hauptzugriff gewähren.
- Ihre Basis-URL MediaTailor für Ihren Quellstandort muss ein v2-Endpunkt sein. MediaPackage
- Der Aufrufer der API muss über `mediapackagev2:GetObject` IAM-Berechtigungen verfügen, um alle Manifeste der obersten Ebene lesen zu können, auf die in den Quellpaketkonfigurationen verwiesen wird. MediaTailor

AWS Secrets Manager Arbeitet mit Zugriffstoken-Authentifizierung

MediaTailor unterstützt die Secrets Manager Manager-Zugriffstoken-Authentifizierung. Bei der AWS Secrets Manager Zugriffstoken-Authentifizierung werden ein AWS Key Management Service (AWS KMS) vom Kunden verwalteter Schlüssel und ein AWS Secrets Manager Geheimnis MediaTailor verwendet, das Sie selbst erstellen, besitzen und verwalten, um Anfragen an Ihren Absender zu authentifizieren.

In diesem Abschnitt erklären wir, wie die Secrets Manager Manager-Zugriffstoken-Authentifizierung funktioniert, und geben step-by-step Informationen zur Konfiguration der Secrets Manager Manager-Zugriffstoken-Authentifizierung. Sie können mit der Secrets Manager Manager-Zugriffstoken-Authentifizierung in AWS Management Console oder programmgesteuert mit arbeiten. AWS APIs

Themen

- [Konfiguration der AWS Secrets Manager Zugriffstoken-Authentifizierung](#)
- [Integration mit MediaPackage Endpunkten, die CDN-Autorisierung verwenden](#)
- [So funktioniert die MediaTailor Secrets Manager Manager-Zugriffstoken-Authentifizierung](#)

Konfiguration der AWS Secrets Manager Zugriffstoken-Authentifizierung

Wenn Sie die AWS Secrets Manager Zugriffstoken-Authentifizierung verwenden möchten, führen Sie die folgenden Schritte aus:

1. Sie [erstellen einen vom AWS Key Management Service Kunden verwalteten Schlüssel](#).
2. Sie [erstellen ein AWS Secrets Manager Geheimnis](#). Das Geheimnis enthält Ihr Zugriffstoken, das in Secrets Manager als verschlüsselter geheimer Wert gespeichert ist. MediaTailor verwendet den vom AWS KMS Kunden verwalteten Schlüssel, um den geheimen Wert zu entschlüsseln.
3. Sie konfigurieren einen AWS Elemental MediaTailor Quellspeicherort für die Verwendung der Secrets Manager Manager-Zugriffstoken-Authentifizierung.

Der folgende Abschnitt enthält step-by-step Anleitungen zur Konfiguration der AWS Secrets Manager Zugriffstoken-Authentifizierung.

Themen

- [Schritt 1: Erstellen Sie einen AWS KMS symmetrischen, vom Kunden verwalteten Schlüssel](#)
- [Schritt 2: Erstellen Sie ein AWS Secrets Manager Geheimnis](#)
- [Schritt 3: Konfigurieren Sie einen MediaTailor Quellstandort mit Zugriffstoken-Authentifizierung](#)

Schritt 1: Erstellen Sie einen AWS KMS symmetrischen, vom Kunden verwalteten Schlüssel

Sie verwenden AWS Secrets Manager , um Ihr Zugriffstoken in Form eines geheimen Codes zu SecretString speichern. Der SecretString wird mithilfe eines AWS KMS symmetrischen, vom Kunden verwalteten Schlüssels verschlüsselt, den Sie erstellen, besitzen und verwalten. MediaTailor

verwendet den symmetrischen, vom Kunden verwalteten Schlüssel, um den Zugriff auf den geheimen Schlüssel zu erleichtern und den geheimen Wert zu verschlüsseln und zu entschlüsseln.

Mit vom Kunden verwalteten Schlüsseln können Sie Aufgaben wie die folgenden ausführen:

- Festlegung und Pflege wichtiger Richtlinien
- Festlegung und Aufrechterhaltung von IAM-Richtlinien und -Zuschüssen
- Aktivieren und Deaktivieren wichtiger Richtlinien
- Rotierendes kryptografisches Schlüsselmateriale
- Hinzufügen von Tags

Informationen zur Verwendung von Secrets Manager AWS KMS zum Schutz von Geheimnissen finden Sie im Thema [AWS Secrets Manager Anwendungsmöglichkeiten AWS KMS](#) im AWS Key Management Service Entwicklerhandbuch.

Weitere Informationen über kundenverwaltete Schlüssel finden Sie unter [Kundenverwaltete Schlüssel](#) im AWS Key Management Service Developer Guide.

 Note

AWS KMS Für die Nutzung eines vom Kunden verwalteten Schlüssels fallen Gebühren an. Weitere Informationen zur Preisgestaltung finden Sie auf der Seite mit den [Preisen für den AWS Key Management Service](#).

Sie können einen AWS KMS symmetrischen, vom Kunden verwalteten Schlüssel mithilfe von AWS Management Console oder programmgesteuert mit dem erstellen. AWS KMS APIs

Einen symmetrischen kundenverwalteten Schlüssel erstellen

Folgen Sie den Schritten zum [Erstellen eines symmetrischen, vom Kunden verwalteten Schlüssels](#) im Entwicklerhandbuch.AWS Key Management Service

Notieren Sie sich den wichtigsten Amazon Resource Name (ARN); Sie benötigen ihn in [Schritt 2: Erstellen Sie ein AWS Secrets Manager Geheimnis](#).

Verschlüsselungskontext

Ein Verschlüsselungskontext ist ein optionaler Satz von Schlüssel-Wert-Paaren, die zusätzliche kontextbezogene Informationen zu den Daten enthalten.

Secrets Manager beinhaltet einen [Verschlüsselungskontext](#) beim Verschlüsseln und Entschlüsseln von `SecretString`. Der Verschlüsselungskontext umfasst den geheimen ARN, der die Verschlüsselung auf dieses spezifische Geheimnis beschränkt. Als zusätzliche Sicherheitsmaßnahme gewährt es MediaTailor in Ihrem Namen einen AWS KMS Zuschuss. MediaTailor wendet eine [GrantConstraints](#) Operation an, die es uns nur ermöglicht, den mit dem geheimen Schlüssel `SecretString` verknüpften ARN zu entschlüsseln, der im Secrets Manager-Verschlüsselungskontext enthalten ist.

Informationen darüber, wie Secrets Manager den Verschlüsselungskontext verwendet, finden Sie im Thema [Verschlüsselungskontext](#) im AWS Key Management Service Entwicklerhandbuch.

Festlegung der Schlüsselrichtlinie

Schlüsselrichtlinien steuern den Zugriff auf den vom Kunden verwalteten Schlüssel. Jeder vom Kunden verwaltete Schlüssel muss über genau eine Schlüsselrichtlinie verfügen, die aussagt, wer den Schlüssel wie verwenden kann. Wenn Sie Ihren vom Kunden verwalteten Schlüssel erstellen, können Sie die Standardschlüsselrichtlinie verwenden. Weitere Informationen finden Sie unter [Authentifizierung und Zugriffskontrolle für AWS KMS](#) im AWS Key Management Service Entwicklerhandbuch.

Um Ihren vom Kunden verwalteten Schlüssel mit Ihren MediaTailor Quellstandortressourcen zu verwenden, müssen Sie dem IAM-Prinzipal, der die folgenden API-Operationen aufruft [CreateSourceLocation](#) oder [UpdateSourceLocation](#) verwendet, die Erlaubnis erteilen:

- `kms:CreateGrant`— Fügt einem vom Kunden verwalteten Schlüssel einen Zuschuss hinzu. MediaTailor erstellt einen Zuschuss für Ihren vom Kunden verwalteten Schlüssel, sodass dieser den Schlüssel verwenden kann, um einen Quellspeicherort zu erstellen oder zu aktualisieren, der mit Zugriffstoken-Authentifizierung konfiguriert ist. Weitere Informationen zur Verwendung von [Grants in AWS KMS](#) finden Sie im AWS Key Management Service Developer Guide.

Auf diese Weise können MediaTailor Sie Folgendes tun:

- Rufen Sie an, `Decrypt` damit Ihr Secrets Manager Manager-Geheimnis beim Aufrufen erfolgreich abgerufen werden kann [GetSecretValue](#).

- Rufen Sie `anRetireGrant`, um den Zuschuss zurückzuziehen, wenn der Quellspeicherort gelöscht oder der Zugriff auf den geheimen Schlüssel gesperrt wurde.

Im Folgenden finden Sie ein Beispiel für eine Richtlinienerklärung, die Sie hinzufügen können: MediaTailor

```
{
  "Sid": "Enable MediaTailor Channel Assembly access token usage for the
MediaTailorManagement IAM role",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::account number:role/MediaTailorManagement"
  },
  "Action": "kms:CreateGrant",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:ViaService": "mediatailor.region.amazonaws.com"
    }
  }
}
```

Weitere Informationen zur Angabe von Berechtigungen in einer Richtlinie und zur Problembehandlung beim Schlüsselzugriff finden Sie unter [Grants in AWS KMS](#) im AWS Key Management Service Developer Guide.

Schritt 2: Erstellen Sie ein AWS Secrets Manager Geheimnis

Verwenden Sie Secrets Manager, um Ihr Zugriffstoken in Form eines durch einen `SecretString` AWS KMS Kunden verwalteten Schlüssels zu speichern. MediaTailor verwendet den Schlüssel, um den zu entschlüsseln. `SecretString` Informationen zur Verwendung von Secrets Manager AWS KMS zum Schutz von Geheimnissen finden Sie im Thema [AWS Secrets Manager Anwendungsmöglichkeiten AWS KMS](#) im AWS Key Management Service Entwicklerhandbuch.

Wenn Sie Origin AWS Elemental MediaPackage als Quelladresse verwenden und die MediaTailor Secrets Manager Manager-Zugriffstoken-Authentifizierung verwenden möchten, gehen Sie wie folgt vor [the section called “Integration mit MediaPackage Endpunkten, die CDN-Autorisierung verwenden”](#).

Sie können ein Secrets Manager-Geheimnis mit dem AWS Management Console oder programmgesteuert mit dem Secrets Manager erstellen. APIs

So erstellen Sie ein Secret

Folgen Sie den Schritten für [AWS Secrets Manager erstellen und verwalten](#) im AWS Secrets Manager Benutzerhandbuch.

Beachten Sie bei der Erstellung Ihres Secrets die folgenden Überlegungen:

- Das `KmsKeyId` muss der [Schlüssel-ARN](#) des vom Kunden verwalteten Schlüssels sein, den Sie in Schritt 1 erstellt haben.
- Sie müssen eine angeben [SecretString](#). Das `SecretString` sollte ein gültiges JSON-Objekt sein, das einen Schlüssel und einen Wert enthält, die das Zugriffstoken enthalten. Zum Beispiel `{"MyAccessTokenIdentifier": "112233445566"}`. Der Wert muss zwischen 8 und 128 Zeichen lang sein.

Wenn Sie Ihren Quellstandort mit Zugriffstoken-Authentifizierung konfigurieren, geben Sie den `SecretString` Schlüssel an. MediaTailor verwendet den Schlüssel, um das in der gespeicherte Zugriffstoken zu suchen und abzurufen `SecretString`.

Notieren Sie sich den geheimen ARN und den `SecretString` Schlüssel. Sie werden sie verwenden, wenn Sie Ihren Quellstandort für die Verwendung der Zugriffstoken-Authentifizierung konfigurieren.

Anhängen einer ressourcenbasierten Geheimrichtlinie

Um MediaTailor auf den geheimen Wert zugreifen zu können, müssen Sie dem Geheimnis eine ressourcenbasierte Richtlinie hinzufügen. Weitere Informationen finden Sie unter [Anhängen einer Berechtigungsrichtlinie an ein AWS Secrets Manager Secret](#) im AWS Secrets Manager Benutzerhandbuch.

Im Folgenden finden Sie ein Beispiel für eine Richtlinienerklärung, die Sie hinzufügen können MediaTailor:

```
{  
  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Principal": {  
        "Service": "mediatailor.amazonaws.com"      }  
    }  
  ]  
}
```

```
    },  
    "Action": "secretsmanager:GetSecretValue",  
    "Resource": "<secret ARN"  
  }  
]  
  
}
```

Schritt 3: Konfigurieren Sie einen MediaTailor Quellstandort mit Zugriffstoken-Authentifizierung

Sie können die Secrets Manager Manager-Zugriffstoken-Authentifizierung mit dem AWS Management Console oder programmgesteuert mit dem konfigurieren. MediaTailor APIs

Um einen Quellstandort mit Secrets Manager Manager-Zugriffstoken-Authentifizierung zu konfigurieren

Folgen Sie den Anweisungen [Access configuration](#) im AWS Elemental MediaTailor Benutzerhandbuch.

Integration mit MediaPackage Endpunkten, die CDN-Autorisierung verwenden

Wenn Sie Origin AWS Elemental MediaPackage als Quellstandort verwenden, MediaTailor kann die Integration mit MediaPackage Endpunkten erfolgen, die CDN-Autorisierung verwenden.

Gehen Sie wie folgt vor, um die Integration mit einem MediaPackage Endpunkt durchzuführen, der die CDN-Autorisierung verwendet.

Zur Integration mit MediaPackage

1. Führen Sie die Schritte unter [CDN-Autorisierung einrichten](#) im AWS Elemental MediaPackage Benutzerhandbuch aus, falls Sie dies noch nicht getan haben.
2. Schließen Sie das Verfahren in a [the section called “Schritt 1: Erstellen Sie einen AWS KMS symmetrischen, vom Kunden verwalteten Schlüssel”](#).
3. Ändern Sie das Geheimnis, das Sie bei der Einrichtung der MediaPackage CDN-Autorisierung erstellt haben. Ändern Sie das Geheimnis mit den folgenden Werten:
 - Aktualisieren Sie den ARN KmsKeyId mit dem vom Kunden verwalteten Schlüssel, in dem Sie ihn erstellt haben[the section called “Schritt 1: Erstellen Sie einen AWS KMS symmetrischen, vom Kunden verwalteten Schlüssel”](#).
 - (Optional) Für können Sie entweder die SecretString UUID auf einen neuen Wert rotieren oder Sie können das vorhandene verschlüsselte Geheimnis verwenden, sofern

es sich um ein Schlüssel- und Wertepaar in einem Standard-JSON-Format handelt, z. B.

```
{"MediaPackageCDNIdentifier": "112233445566778899"}
```

4. Führen Sie die Schritte unter [au the section called "Anhängen einer ressourcenbasierten Geheimrichtlinie"](#).
5. Führen Sie die Schritte unter [au the section called "Schritt 3: Konfigurieren Sie einen MediaTailor Quellstandort mit Zugriffstoken-Authentifizierung"](#).

So funktioniert die MediaTailor Secrets Manager Manager-Zugriffstoken-Authentifizierung

Nachdem Sie einen Quellspeicherort für die Verwendung der Zugriffstoken-Authentifizierung erstellt oder aktualisiert haben, MediaTailor fügt er das Zugriffstoken in einen HTTP-Header ein, wenn Sie Quellinhaltsmanifeste von Ihrem Ursprung anfordern.

Im Folgenden finden Sie eine Übersicht darüber, wie die Secrets Manager Manager-Zugriffstoken-Authentifizierung für die Herkunftsauthentifizierung am Quellstandort MediaTailor verwendet wird:

1. Wenn Sie einen MediaTailor Quellspeicherort erstellen oder aktualisieren, der die Zugriffstoken-Authentifizierung verwendet, MediaTailor sendet eine [DescribeSecret](#)Anfrage an Secrets Manager, um den mit dem Secret verknüpften AWS KMS Schlüssel zu ermitteln. Sie nehmen den geheimen ARN in Ihre Konfiguration für den Zugriff auf den Quellstandort auf.
2. MediaTailor erstellt eine Genehmigung für den vom Kunden verwalteten Schlüssel, MediaTailor sodass dieser mithilfe des Schlüssels auf das in der gespeicherte [Zugriffstoken](#) zugreifen und es entschlüsseln kann. SecretString Der Grant-Name wird lauten. MediaTailor-SourceLocation-*your AWS-Konto ID-source location name*

Sie können den Zugriff auf den Zuschuss jederzeit widerrufen oder ihm den Zugriff auf den vom Kunden verwalteten Schlüssel entziehen MediaTailor. Weitere Informationen finden Sie unter [RevokeGrant](#) in der AWS Key Management Service -API-Referenz.

3. Wenn eine VOD-Quelle erstellt, aktualisiert oder in einem Programm verwendet wird, sendet sie HTTP-Anfragen an MediaTailor die Quellverzeichnisse, um die Quellinhaltsmanifeste abzurufen, die den VOD-Quellen am Quellspeicherort zugeordnet sind. Wenn die VOD-Quelle einem Quellstandort zugeordnet ist, für den ein Zugriffstoken konfiguriert ist, enthalten die Anfragen das Zugriffstoken als HTTP-Header-Wert.

Mit VOD-Quellen arbeiten

Eine VOD-Quelle steht für einen einzelnen Inhalt, z. B. ein Video oder eine Episode eines Podcasts, den Sie zu Ihrem Quellstandort hinzufügen. Du fügst deinem Quellort eine oder mehrere VOD-Quellen hinzu und ordnest dann jede VOD-Quelle einem Programm zu, nachdem du deinen Kanal erstellt hast.

Jede VOD-Quelle muss mindestens eine Paketkonfiguration haben. Eine Paketkonfiguration gibt ein Paketformat, einen Manifestspeicherort und eine Quellgruppe für Ihre VOD-Quelle an. Wenn Sie Ihren Kanal erstellen, verwenden Sie die Quellgruppen der Paketkonfiguration, um die entsprechenden Ausgaben auf Ihrem Kanal zu erstellen. Wenn eine Quelle beispielsweise in zwei verschiedenen Formaten — HLS und DASH — verpackt ist, würdest du zwei Paketkonfigurationen erstellen, eine für DASH und eine für HLS. Dann würden Sie zwei Kanalausgänge erstellen, einen für jede Paketkonfiguration. Jeder Kanalausgang bietet einen Endpunkt, der für Wiedergabeanfragen verwendet wird. Im obigen Beispiel würde der Kanal also einen Endpunkt für HLS-Wiedergabeanfragen und einen Endpunkt für DASH-Wiedergabeanfragen bereitstellen.

Wenn Sie möchten, dass die Offsets der Werbemarkierungen in Ihrem Manifest automatisch erkannt werden, muss jede Werbemarkierung in allen Paketkonfigurationen mit demselben Offset erscheinen und eine Dauer von Null haben. MediaTailor wird bei HLS erkannt DATERANGE und EXT-X-CUE-OUT markiert. Bei DASH erkennt HLS das erste Event-Tag innerhalb jedes EventStream Tags.

Im folgenden Beispiel wird eine Werbeunterbrechungsmöglichkeit aufgrund des DATERANGE Tags mit einer Dauer von 0,0 bei einem Offset von 12000 ms erkannt. Das erste DATERANGE Tag mit einem Offset von 0 ms wird nicht erkannt, da es eine Dauer von 10,0 hat.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:6
#EXT-X-MEDIA-SEQUENCE:0
#EXT-X-DATERANGE:ID="1001",START-DATE="2021-09-16T23:51:05.249Z",DURATION=10.0,SCTE35-
OUT=0xFC302500000003289800FFF01405000003E97FEFFE1D381BD8FE000DBBA00001010100000FD2B275
#EXTINF:6.000,
../.../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_0.ts
#EXTINF:6.000,
../.../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_1.ts
#EXT-X-DATERANGE:ID="1001",START-DATE="2021-09-16T23:51:05.249Z",DURATION=0.0,SCTE35-
OUT=0xFC302500000003289800FFF01405000003E97FEFFE1D381BD8FE000DBBA00001010100000FD2B275
#EXTINF:6.000,
../.../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_2.ts
```

Im folgenden Beispiel wird eine Werbeblock-Opportunity mit einem Offset von 0 ms erkannt, da das EXT-X-CUE-OUT Tag eine Dauer von 0 hat und unmittelbar darauf ein EXT-X-CUE-IN Tag folgt. Das zweite EXT-X-CUE-IN Paar EXT-X-CUE-OUT/wird nicht erkannt, da es eine Dauer von 10 hat.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-TARGETDURATION:6
#EXT-X-MEDIA-SEQUENCE:0
#EXT-X-CUE-OUT:0
#EXT-X-CUE-IN
#EXTINF:6.000,
../../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_0.ts
#EXTINF:6.000,
../../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_1.ts
#EXT-X-CUE-OUT:10
...
#EXT-X-CUE-IN
#EXTINF:6.000,
../../719f911124e0495cbb067c91c1d6c298/1785a16ca14d4c2884781f25333f6766/index_1_2.ts
```

Im folgenden Beispiel wird eine Werbeunterbrechungsmöglichkeit mit einem Offset von 0 ms erkannt, da das erste Ereignis in dem Zeitraum EventStream eintritt, der bei PT0.000S beginnt. Das zweite Ereignis Event im EventStream wird nicht erkannt.

```
<Period start="PT0.000S" id="9912561" duration="PT29.433S">
<EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
<Event duration="0">
  <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="1241950593" tier="4095">
    <scte35:SpliceInsert spliceEventId="99" spliceEventCancelIndicator="false"
outOfNetworkIndicator="true" spliceImmediateFlag="false" uniqueProgramId="1"
availNum="1" availsExpected="1">
      <scte35:Program><scte35:SpliceTime ptsTime="3552273000"/></scte35:Program>
      <scte35:BreakDuration autoReturn="true" duration="2700000"/>
    </scte35:SpliceInsert>
  </scte35:SpliceInfoSection>
</Event>
<Event duration="0">
  <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="1241950593" tier="4095">
    <scte35:SpliceInsert spliceEventId="99" spliceEventCancelIndicator="false"
outOfNetworkIndicator="true" spliceImmediateFlag="false" uniqueProgramId="1"
availNum="1" availsExpected="1">
      <scte35:Program><scte35:SpliceTime ptsTime="3552273000"/></scte35:Program>
```

```
<scte35:BreakDuration autoReturn="true" duration="2700000"/>
</scte35:SpliceInsert>
</scte35:SpliceInfoSection>
</Event>
</EventStream>
...
</Period>
```

Hinzufügen von VOD-Quellen zu Ihrem Quellstandort

Im folgenden Verfahren wird erklärt, wie Sie VOD-Quellen zu Ihrem Quellspeicherort hinzufügen und Paketkonfigurationen mithilfe der MediaTailor Konsole einrichten. Informationen zum Hinzufügen von VOD-Quellen mithilfe der MediaTailor API finden Sie [CreateVodSource](#) in der AWS Elemental MediaTailor API-Referenz.

Important

Bevor Sie Ihre VOD-Quellen hinzufügen, stellen Sie sicher, dass sie die folgenden Anforderungen erfüllen:

- Die Quellvarianten müssen alle dieselbe Länge haben, wie im Quellmanifest festgelegt.
- Innerhalb einer Paketkonfiguration muss jede Quelle dieselbe Anzahl von untergeordneten Streams haben.

Aufgrund dieser Anforderungen unterstützen wir weder pro Titel noch automatisiertes ABR, da diese Kodierungsmethoden unterschiedliche Manifestlängen und untergeordnete Streams erzeugen können.

Wir empfehlen Ihnen, eine Kodierungsvorlage zu verwenden, die eine Mindestsegmentlänge beinhaltet, um sicherzustellen, dass Ihre codierten Quellen diese Anforderungen erfüllen.

Um VOD-Quellen zu Ihren Quellspeicherorten hinzuzufügen

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>
2. Wählen Sie im Navigationsbereich Channel-Assembly > Source Locations aus.
3. Wählen Sie im Bereich Quellverzeichnisse den Quellspeicherort aus, den Sie in dem [So erstellen Sie einen Quellspeicherort](#) Verfahren erstellt haben.
4. Wählen Sie VOD-Quelle hinzufügen aus.

5. Geben Sie unter VOD-Quelldetails einen Namen für Ihre VOD-Quelle ein:
 - Name: Eine Kennung für Ihre VOD-Quelle, z. B. my-example-video
6. **source-group-name** Geben Sie unter Paketkonfigurationen > Informationen zur Paketkonfiguration ein:

 Note

Die Paketkonfigurationen Ihrer Quelle müssen alle dieselbe Dauer haben, wie im Manifest der Quelle festgelegt. Und alle Quellen innerhalb einer Paketkonfiguration müssen dieselbe Anzahl von untergeordneten Streams haben. Um diese Anforderungen zu erfüllen, empfehlen wir Ihnen, eine Kodierungsvorlage für Ihre Assets zu verwenden. Wir empfehlen, eine Kodierungsvorlage mit einer Mindestsegmentlänge von einer Sekunde zu verwenden. MediaTailor unterstützt kein Streaming pro Titel oder automatisiertes Streaming mit adaptiver Bitrate (ABR), da diese Kodierungsmethoden gegen diese Anforderungen verstoßen.

- Quellgruppe: Geben Sie einen Quellgruppennamen ein, der diese Paketkonfiguration beschreibt, z. B. HLS-4k. Notieren Sie sich diesen Namen. Sie werden ihn angeben, wenn Sie die Ausgabe Ihres Kanals erstellen. Weitere Informationen finden Sie unter [Verwende Quellgruppen mit den Ausgaben deines Kanals](#).
- Typ: Wählen Sie das Paketformat für diese Konfiguration aus. MediaTailor unterstützt HLS und DASH.
- Relativer Pfad: Der relative Pfad von der Basis-HTTP-URL des Quellspeicherorts zum Manifest. Zum Beispiel /my/path/index.m3u8.

 Note

MediaTailor importiert automatisch alle Untertitel und untergeordneten Streams, die in einem übergeordneten Manifest enthalten sind. Sie müssen nicht für jede Ihrer Quell-Wiedergabeverversionen (DASH) oder Variant-Streams (HLS) separate Paketkonfigurationen erstellen.

Weitere Informationen zu Paketkonfigurationen finden Sie unter. [Verwenden von Paketkonfigurationen](#)

7. Wählen Sie VOD-Quelle hinzufügen.

Wenn Sie weitere VOD-Quellen hinzufügen möchten, wiederholen Sie die Schritte 4-7 des Verfahrens.

Arbeiten mit Live-Quellen

Eine Live-Quelle steht für einen einzelnen Live-Stream, z. B. ein Live-Fußballspiel oder eine Nachrichtenübertragung, den Sie zu Ihrem Quellstandort hinzufügen. Nachdem du deinen Kanal erstellt hast, fügst du deinem Quellstandort eine oder mehrere Live-Quellen hinzu und ordnest dann jede Live-Quelle einem Programm zu.

MediaTailor unterstützt die folgenden Arten der linearen Kanalanordnung:

- VOD-Quellen für einen Kanal, der Inhalte enthält VOD-to-live
- Live-Quellen für einen Kanal, der live-to-live Inhalte enthält, die mit Inhalten vermischt sind VOD-to-live

Ein Beispiel für VOD-to-live Inhalte ist ein Kanal, der eine Bibliothek mit VOD-Assets zu einem Live-Stream zusammenfügt. Ein Beispiel für live-to-live Inhalte, die mit VOD-to-live Inhalten gemischt werden, ist ein Kanal, auf dem hauptsächlich VOD-Inhalte gezeigt werden, mit Ausnahme einer nächtlichen Nachrichtenveranstaltung oder einer vorab geplanten Live-Sportveranstaltung. Ein anderes Beispiel für live-to-live Inhalte, die mit VOD-to-live Inhalten gemischt werden, ist ein reiner live-to-live Kanal, dessen Herkunft je nach Tageszeit variiert.

Sie können Live-Quellen verwenden, um einen regionalen Kanal einzurichten, der hauptsächlich nationale Programme zeigt, aber auch regionale Programmüberschreibungen beinhaltet und VOD-Inhalte enthält. Um das zu tun, müssen Sie einen Preis zahlen. `encoder/packager pair for the national content, then run regional encoders when those regions are live. Then, you create regional channel-assembly channels, each with their own schedules. This way, viewers can switch back and forth as needed. This setup helps you minimize encoding/packaging`

Jede Live-Quelle muss mindestens eine Paketkonfiguration haben. Eine Paketkonfiguration gibt ein Paketformat, einen Speicherort für das Manifest und eine Quellgruppe für Ihre Live-Quelle an. Wenn Sie Ihren Kanal erstellen, verwenden Sie die Quellgruppen der Paketkonfiguration, um die entsprechenden Ausgaben auf Ihrem Kanal zu erstellen. Wenn eine Quelle beispielsweise in zwei verschiedenen Formaten — HLS und DASH — verpackt ist, würdest du zwei Paketkonfigurationen erstellen, eine für DASH und eine für HLS. Dann würden Sie zwei Kanalausgänge erstellen, einen

für jede Paketkonfiguration. Jeder Kanalausgang bietet einen Endpunkt, der für Wiedergabeanfragen verwendet wird. In diesem Beispiel bietet der Kanal einen Endpunkt für HLS-Wiedergabeanfragen und einen Endpunkt für DASH-Wiedergabeanfragen.

Allgemeine Anforderungen für die Verwendung von Live-Quellen

Wenn Sie Live-Quellen verwenden, müssen Ihre Inhalte den folgenden allgemeinen Anforderungen entsprechen:

- **HLS-Live-Quellen** — Sie müssen `#EXT-X-PROGRAM-DATE-TIME` Tags für das erste Segment im Manifestfenster und für jede Unterbrechung angeben.
- **HLS** — Sie müssen Anzeigenmarkierungen als konfigurieren. `DATERANGE`
- **Fenster mit Quellmanifest** — Wir empfehlen, ein Manifestfenster mit einer Dauer zu verwenden, die mindestens so lang ist wie das Manifestfenster auf Ihrem MediaTailor Channel Assembly-Kanal. Es hat sich bewährt, ein Manifestfenster mit einer Dauer von 30 Sekunden oder länger als das Manifestfenster auf dem Channel Assembly-Kanal zu verwenden.
- **Passen Sie die Zieldauer an die Dauer der vorhandenen Quellen an.**
- **Passen Sie die Anzahl der untergeordneten Playlisten an die Anzahl der vorhandenen Quellen an.**

Konfigurationen

Wenn Sie andere Mediendienste von AWS Elemental als Teil Ihres Workflows für Live-Quellen verwenden, empfehlen wir, bei der Einrichtung Ihrer MediaPackage Konfiguration die bewährten Methoden zu befolgen. In der folgenden Tabelle wird beschrieben, wie Sie MediaPackage Einstellungen auf der Grundlage des von Ihnen verwendeten Streaming-Standards konfigurieren.

MediaPackage Einrichtung für Live-Quellen

Standard	Einstellung	Wert	Notwendigkeit	Hinweise
HLS	Endpunkttyp	Apple HLS	Erforderlich, sofern nicht CMAF verwendet wird	Um HLS-Jobs zuzuordnen ts AWS Elemental MediaConvert
HLS	Endpunkttyp	CMAF	Erforderlich, sofern Sie	Um mp4 AWS Elemental MediaConv

Standard	Einstellung	Wert	Notwendigkeit	Hinweise
			Apple HLS nicht verwenden	ert HLS-Jobs abzugleichen
HLS	ProgramDateTimeIntervalSeconds	1	Erforderlich	Sie müssen für jedes Segment einen Wert angeben#EXT-X-PROGRAM-DATE-TIME , um Wiedergabeprobleme bei Unterbrechungen zu vermeiden.
HLS	PlaylistWindowSeconds	30 Sekunden länger als das Channel-Assembly-Manifestfenster	Erforderlich	
HLS	AdMarkers	DATERANGE	Erforderlich beim Durchgeben von Werbemarkierungen	
HLS	IncludeIframeOnlyStream	Disabled	Empfohlen	
DASH	ManifestLayout	FULL	Empfohlen	

Standard	Einstellung	Wert	Notwendigkeit	Hinweise
DASH	SegmentTemplateFormat	NUMBER_WITH_TIMELINE oder TIME_WITH_TIMELINE	Empfohlen	NUMBER_WITH_DURATION wird nicht unterstützt.
DASH	ManifestWindowSeconds	30 Sekunden länger als das Channel-Assembly-Manifestfenster	Erforderlich	
DASH	PeriodTriggers	ADS	Erforderlich beim Durchgeben von Werbemarkierungen	

Live-Quellen zu Ihrem Quellstandort hinzufügen

Im folgenden Verfahren wird erklärt, wie Sie mit der MediaTailor Konsole Live-Quellen zu Ihrem Quellstandort hinzufügen und Paketkonfigurationen einrichten. Informationen zum Hinzufügen von Live-Quellen mithilfe der MediaTailor API finden Sie [CreateLiveSource](#) in der AWS Elemental MediaTailor API-Referenz.

Important

Bevor Sie Ihre Live-Quellen hinzufügen, stellen Sie sicher, dass innerhalb einer Paketkonfiguration jede Quelle dieselbe Anzahl von untergeordneten Streams hat.

Um Live-Quellen zu Ihren Quellspeicherorten hinzuzufügen

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Source Locations aus.

3. Wählen Sie im Bereich Quellverzeichnis den Quellspeicherort aus, den Sie in dem [So erstellen Sie einen Quellspeicherort](#) Verfahren erstellt haben.
4. Wählen Sie auf der Registerkarte Live-Quellen die Option Live-Quelle hinzufügen aus.
5. Geben Sie unter Live-Quelldetails einen Namen für Ihre Live-Quelle ein:
 - Name: Eine Kennung für Ihre Live-Quelle, z. my-example-videoB.
6. **source-group-name** Geben Sie unter Paketkonfigurationen > Informationen zur Paketkonfiguration ein:

 Note

Innerhalb einer Paketkonfiguration müssen alle VOD-Quellen und Live-Quellen dieselbe Anzahl an untergeordneten Streams haben. Wir empfehlen, dass Sie Ihre Quell-Streams auf die gleiche Weise konfigurieren.

- Quellgruppe: Geben Sie einen Quellgruppennamen ein, der diese Paketkonfiguration beschreibt, z. B. HLS-4k. Notieren Sie sich diesen Namen. Sie werden ihn angeben, wenn Sie die Ausgabe Ihres Kanals erstellen. Weitere Informationen finden Sie unter [Verwenden von Quellgruppen mit den Ausgaben deines Kanals](#).
- Typ: Wählen Sie das Paketformat für diese Konfiguration aus. MediaTailor unterstützt HLS und DASH.
- Relativer Pfad: Der relative Pfad von der Basis-HTTP-URL des Quellspeicherorts zum Manifest. Zum Beispiel/my/path/index.m3u8.

 Note

MediaTailor importiert automatisch alle Untertitel und untergeordneten Streams, die in einem übergeordneten Manifest enthalten sind. Sie müssen nicht für jede Ihrer Quell-Wiedergabeversionen (DASH) oder Variant-Streams (HLS) separate Paketkonfigurationen erstellen.

Weitere Informationen zu Paketkonfigurationen finden Sie unter. [Verwenden von Paketkonfigurationen](#)

7. Wählen Sie Live-Quelle hinzufügen.

Wenn Sie weitere Live-Quellen hinzufügen möchten, wiederholen Sie die Schritte 4-6 des Verfahrens.

Verwenden von Paketkonfigurationen

Eine Paketkonfiguration ist eine Darstellung der Quelle, die die verschiedenen Verpackungsmerkmale enthält, die für die Wiedergabe auf verschiedenen Geräten erforderlich sind. Angenommen, Sie haben eine Quelle mit drei Paketformaten: HLS mit DRM, DASH mit Segment-Timeline-Adressierung und HLS mit CMAF-Segmenten.

Bei der Channel-Assembly werden Ihre Quellen nicht neu verpackt. Wenn Sie mehrere Paketformate für eine bestimmte Quelle einbeziehen möchten, müssen Sie jedes Paketformat am Quellspeicherort verfügbar machen und den Pfad zu jedem Paketformat angeben.

Jedes Paketkonfigurationsobjekt muss Folgendes enthalten:

- Relativer Pfad — Der vollständige Pfad zum Paketformat der Quelle, relativ zum Quellspeicherort. Zum Beispiel `/my/path/index.m3u8`.
- Quellgruppe — Der Name der Quellgruppe, die verwendet wird, um Paketkonfigurationen mit der Ausgabe eines Kanals zu verknüpfen.
- Typ — Entweder HLS oder DASH.

Nachdem Sie einen Kanal erstellt haben, müssen Sie auch jede Quellgruppe deklarieren, die Sie für die Ausgabe des Kanals verwenden möchten.

Zwischenspeichern von Manifesten

MediaTailor speichert die Quell-Playlisten regelmäßig und bei Gelegenheit im Cache, um die Leistung und Zuverlässigkeit der Kanalmontage zu verbessern. Manchmal ist die zwischengespeicherte Version im Vergleich zur Originalversion an Ihrem Quellspeicherort veraltet. Rufen Sie auf MediaTailor, um die Aktualisierung der zwischengespeicherten Version der Quelle zu erzwingen. [UpdateVodSource](#) Verwenden Sie diesen Aufruf beispielsweise, wenn sich die eingebetteten Pfade in Ihrer Quelle ändern. Stellen Sie sicher, dass Sie immer eine up-to-date Version der Quelle an Ihrem Quellspeicherort verfügbar haben, auch wenn Sie nur wenige Anfragen von erhalten MediaTailor.

Mit Kanälen arbeiten

Ein Kanal fügt Ihre Quellmanifeste zu einem linearen Stream zusammen. Jeder Kanal enthält einen oder mehrere Ausgänge, die Ihren Paketkonfigurationen entsprechen.

Zuerst erstellen Sie einen Kanal und fügen dann Ihre VOD-Quellen und Live-Quellen zum Zeitplan des Kanals hinzu, indem Sie Programme erstellen. Jedes Programm ist mit einer VOD-Quelle oder einer Live-Quelle verknüpft.

Themen

- [Erstellen Sie mit der MediaTailor Konsole einen Kanal](#)
- [Verwende Quellgruppen mit den Ausgaben deines Kanals](#)
- [Löschen Sie einen Kanal mithilfe der Konsole MediaTailor](#)

Erstellen Sie mit der MediaTailor Konsole einen Kanal

Im folgenden Verfahren wird beschrieben, wie Sie mithilfe der MediaTailor Konsole einen Kanal erstellen.

So erstellen Sie einen Channel

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie in der Navigationsleiste die Option Kanal erstellen aus.
4. Geb unter Kanaldetails Details zu deinem Kanal ein:
 - Name: Gib einen Namen für deinen Kanal ein.
 - Stufe: Die Stufe bestimmt, welche Funktionen der Kanal unterstützt und wie viel der Betrieb des Kanals kostet. Weitere Informationen zur Preisgestaltung finden Sie auf der [Preisseite von Channel Assembly](#). MediaTailor unterstützt die folgenden Stufen:
 - Basic — Die Basic-Stufe unterstützt sowohl den linearen als auch den Loop-Wiedergabemodus und unterstützt keine Live-Quellen.
 - Standard — Die Standard-Stufe unterstützt Live-Quellen und erfordert den linearen Wiedergabemodus.

Wenn Sie in den Kanaldetails die Option Standard auswählen, können Sie die Zielgruppen unter Zielgruppendedetails definieren. Diese Zielgruppen werden für ProgramRules verwendet, wenn Sie AudienceMedia für Ihr Standardprogramm erstellen.

- Wählen Sie Hinzufügen aus.
 - Geben Sie den Namen der Zielgruppe in das Textfeld ein. Er muss zwischen 1 und 32 alphanumerische Zeichen lang sein.
 - Wählen Sie Bestätigen aus.
 - Wählen Sie Weiter.
- Wiedergabemodus: Der Wiedergabemodus legt das Wiedergabeverhalten des Kanals fest. MediaTailor unterstützt die folgenden Wiedergabemodi:
- Loop — Die Programme im Zeitplan werden back-to-back in einer Endlosschleife abgespielt. Nachdem das letzte Programm in einem Zeitplan abgespielt wurde, kehrt die Wiedergabe zum ersten Programm zurück. Die Wiedergabe wird in einer Endlosschleife fortgesetzt, bis Sie den Kanal beenden.
 - Linear — Jedes Programm im Zeitplan wird einmal abgespielt, back-to-back.

5.

Wählen Sie für Filler Slate den Quellspeicherort, der auf den Slate-Speicherort verweist, und den Namen der VOD-Quelle, die als Slate verwendet werden soll. MediaTailor verwendet das Slate, um Lücken zwischen Programmen im Zeitplan zu füllen. Wenn die Dauer der Slate kürzer als die Dauer der Lücke zwischen den Programmen ist, wird die MediaTailor Slate wiederholt. Sie müssen das Feld Filler Slate konfigurieren, wenn Ihr Kanal den linearen Wiedergabemodus verwendet. MediaTailor unterstützt Filler Slate für den Loop-Wiedergabemodus nicht.

6. Wählen Sie Weiter.

7. Geben Sie die Zielgruppendedetails in den Programmregeln an.

8. Wenn Sie in den Kanaldetails die Option Standard auswählen, können Sie die Zielgruppen unter Zielgruppendedetails definieren. Diese Zielgruppen werden für Programmregeln verwendet, wenn Sie AudienceMedia für Ihr Standardprogramm erstellen:

- Wählen Sie Hinzufügen aus, fügen Sie im Textfeld eine Zielgruppe hinzu und wählen Sie dann Bestätigen aus.

 Note

Geben Sie einen Namen ein, der nicht länger als _ 32 alphanumerische Zeichen ist.

- **Ausgabebetyp:** Wählen Sie das Streaming-Format für den Kanal aus. DASH und HLS werden unterstützt.
 - **Quellgruppe:** Geben Sie den Namen der Quellgruppe ein, die Sie in Ihrer Paketkonfiguration erstellt haben, wie unter beschrieben. [Hinzufügen von VOD-Quellen zu Ihrem Quellstandort](#)
9. Klicken Sie auf Weiter.
10. Geben Sie unter Manifesteinstellungen zusätzliche Informationen zu Ihren Manifesteinstellungen ein:
- **Manifestfenster (Sekunden):** Das Zeitfenster in Sekunden, das in jedem Manifest enthalten ist. Der Mindestwert ist 30 Sekunden und der Höchstwert ist 3600 Sekunden.
 - **Art der Anzeigenmarkierung (nur HLS-Ausgaben):** Die Art der Anzeigen-Tags, die in Werbeunterbrechungen von VOD-Programmen erscheinen. Wählen Sie `Daterange` diese Option aus, um Werbeunterbrechungen mit Tags in VOD-Programme MediaTailor einzufügen. `EXT-X-DATERANGE` Wählen Sie `Scte35 Enhanced` diese Option aus, um Werbeunterbrechungen mithilfe `EXT-X-CUE-OUT` von `UND`-Tags in VOD-Programme MediaTailor einzufügen. `EXT-X-CUE-IN` Weitere Informationen zu diesen Tag-Typen finden Sie unter [SCTE-35-Nachrichten für Werbeunterbrechungen](#). Leitet bei Live-Workflows MediaTailor immer `DATERANGE` Tags und keine erweiterten `Scte35`-Tags weiter, unabhängig vom ausgewählten Anzeigen-Markup-Typ.
11. Wenn Sie mehrere Kanalausgänge konfigurieren möchten, wählen Sie unter Ausgänge die Option Hinzufügen. Konfigurieren Sie dann die Details für Ihre Ausgabe, indem Sie die Schritte 6 und 7 in diesem Verfahren ausführen.
12. Wählen Sie Weiter.
13. Wählen Sie unter Kanalrichtlinie die IAM-Richtlinieneinstellungen Ihres Kanals aus:
- **Keine Kanalrichtlinie anhängen:** Beschränken Sie die Wiedergabe nur auf Personen, die Zugriff auf die Anmeldeinformationen dieses Kontos haben.
 - **Benutzerdefinierte Richtlinie anhängen:** Definieren Sie Ihre eigene Richtlinie und beschränken Sie den Zugriff auf so wenige oder so viele, wie Sie möchten.
 - **Öffentliche Richtlinie anhängen:** Akzeptiere alle eingehenden Kundenanfragen an die Ausgabe eines Kanals. Sie müssen diese Option verwenden, wenn Sie die MediaTailor Anzeigeneinfügung verwenden möchten.
14. Wählen Sie Weiter.
15. Überprüfen Sie Ihre Einstellungen im Bereich Überprüfen und erstellen.

16. Wählen Sie Create channel (Channel erstellen).

Note

Kanäle werden im Status „Gestoppt“ erstellt. Dein Kanal ist erst aktiv, wenn du ihn mit der MediaTailor Konsole oder der MediaTailor StartChannel API startest.

Verwende Quellgruppen mit den Ausgaben deines Kanals

Eine Quellgruppe verknüpft eine Paketkonfiguration mit einer Ausgabe auf einem Kanal. Wenn Sie die Paketkonfiguration auf der Quelle erstellen, identifizieren Sie den Namen der Quellgruppe. Wenn Sie dann die Ausgabe auf dem Kanal erstellen, geben Sie denselben Namen ein, um die Ausgabe mit der Paketkonfiguration zu verknüpfen. VOD-Quellen und Live-Quellen, die einem Programm auf einem Kanal hinzugefügt werden, müssen zu der Quellgruppe gehören, die in der Ausgabe angegeben ist.

Zum Beispiel:

- Die VOD-Quellen 1 und 2 haben beide drei Paketkonfigurationen mit den Quellgruppen HLS, DASH und HLS-4k.
- VOD-Quelle 3 hat zwei Paketkonfigurationen mit den Quellgruppen HLS und DASH.

Wenn Kanal A zwei Ausgänge mit den Quellgruppen HLS und DASH hat, kann der Kanalausgang alle drei VOD-Quellen verwenden. Das liegt daran, dass die VOD-Quellen 1, 2 und 3 alle Paketkonfigurationen mit den Quellgruppenbezeichnungen HLS und DASH haben.

Wenn Kanal B zwei Ausgänge mit den Quellgruppen HLS und HLS-4k hat, kann er die VOD-Quelle 1 und 2 verwenden, aber nicht 3. Das liegt daran, dass die VOD-Quellen 1 und 2 beide Paketkonfigurationen mit den Quellgruppenbezeichnungen HLS und HLS-4k haben.

Wenn Kanal C einen einzigen Ausgang mit der Quellgruppe DASH hat, kann er alle drei VOD-Quellen verwenden. Alle drei VOD-Quellen haben Paketkonfigurationen mit der DASH-Quellgruppe.

Löschen Sie einen Kanal mithilfe der Konsole MediaTailor

Gehen Sie wie folgt vor, um Ihren Kanal zu löschen.

Um deinen Kanal zu löschen

1. Öffne die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie den Kanal aus, den Sie löschen möchten.
4. Wenn dein Kanal läuft, wähle im Drop-down-Menü Aktionen die Option Stopp aus. Du musst deinen Kanal beenden, bevor du ihn löschen kannst.
5. Wenn dein Kanal gestoppt ist, wähle im Drop-down-Menü „Aktionen“ die Option Löschen aus.

Ein Programm zum Zeitplan eines Kanals hinzufügen

Jedes Programm enthält eine VOD-Quelle oder eine Live-Quelle, die Teil eines Quellenverzeichnisses in Ihrem Konto ist. Du fügst deine Programme zum Zeitplan deines Kanals hinzu, um die Reihenfolge zu kontrollieren, in der sie im Stream deines Kanals abgespielt werden.

Ein Programm, das eine VOD-Quelle enthält, kann mit einer oder mehreren Werbeunterbrechungen konfiguriert werden. Jede Werbeunterbrechung enthält einen Slate, bei dem es sich um eine VOD-Quelle von einem Quellort handelt. Um den Werbeblock zu erstellen, fügen Sie den Slate mit einem Abstand von Millisekunden in das Programm ein.

Themen

- [Ein Programm innerhalb eines Kanalzeitplans mithilfe der Konsole erstellen MediaTailor](#)
- [Definition von Zielgruppenkohorten und alternativen Inhalten mithilfe der Programmregeln](#)
- [Generierung zielgruppenspezifischer Manifeste](#)

Ein Programm innerhalb eines Kanalzeitplans mithilfe der Konsole erstellen MediaTailor

Im folgenden Verfahren wird beschrieben, wie Sie mithilfe der MediaTailor Konsole ein Programm innerhalb des Zeitplans Ihres Kanals erstellen. Außerdem wird beschrieben, wie Werbeunterbrechungen konfiguriert werden, die optional sind. Informationen zum Erstellen von Programmen mithilfe der MediaTailor API finden Sie [CreateProgram](#) in der AWS Elemental MediaTailor API-Referenz.

Um ein Programm hinzuzufügen

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie im Bereich Kanäle den Kanal aus, den Sie in dem [So erstellen Sie einen Channel](#) Verfahren erstellt haben.
4. Geben Sie in den Programmdetails Details zu Ihrem Programm ein:
 - Name: Dies ist der Name des Programms, das du deinem Kanal hinzufügst.
 - Quelltyp: Legt fest, welche Art von Quellvideo das Programm abspielt. Diese Option ist nur für Standardkanäle verfügbar.
 - VOD — Das Programm spielt eine video-on-demand Quelle ab, z. B. eine zuvor aufgezeichnete TV-Episode.
 - Live — Das Programm spielt eine Live-Quelle ab, z. B. eine Live-Nachrichtenübertragung.
 - Name des Quellverzeichnisses: Der Quellspeicherort, MediaTailor der dem Programm zugeordnet ist.
 - Wenn Sie „Bestehenden Quellspeicherort auswählen“ wählen, wählen Sie im Menü „Quellspeicherort auswählen“ einen Quellspeicherort aus. Sie können auch anhand des Namens nach Ihrem Quellstandort suchen. Dies ist hilfreich, wenn Sie über eine große Anzahl von Quellstandorten verfügen.
 - Wenn Sie Geben Sie den Quellspeicherort ein wählen, suchen Sie anhand des Namens nach Ihrem Quellspeicherort.
 - Name der VOD-Quelle: Der Name der VOD-Quelle, die dem Programm MediaTailor zugeordnet ist:
 - Wenn Sie Select an existing VOD-Quelle wählen, wählen Sie einen VOD-Quellennamen aus der Liste der VOD-Quellen, die Ihrem Konto zugeordnet sind. Sie können auch anhand des Namens nach Ihrer VOD-Quelle suchen. Dies ist hilfreich, wenn Sie über eine große Anzahl von VOD-Quellen verfügen.
 - Wenn Sie „Nach Namen suchen“ wählen, suchen Sie anhand des Namens nach Ihrer Live-Quelle.
 - Name der Live-Quelle: Der Name der Live-Quelle, die dem Programm zugeordnet werden soll. Diese Option ist nur verfügbar, wenn Sie Live als Quelltyp ausgewählt haben.
 - Wenn Sie „Bestehenden Quellspeicherort auswählen“ wählen, wählen Sie im Menü „Quellspeicherort auswählen“ einen Quellspeicherort aus. Sie können auch anhand des

Namens nach Ihrem Quellstandort suchen. Dies ist hilfreich, wenn Sie über eine große Anzahl von Quellstandorten verfügen.

- Wenn Sie Geben Sie den Quellspeicherort ein wählen, suchen Sie anhand des Namens nach Ihrem Quellspeicherort.
- Name der VOD-Quelle: Der Name der VOD-Quelle, die dem Programm MediaTailor zugeordnet ist:
 - Wenn Sie „Vorhandene Live-Quelle auswählen“ wählen, wählen Sie einen Namen für die Live-Quelle aus der Liste der Live-Quellen aus, die mit Ihrem Konto verknüpft sind. Sie können Ihre Live-Quelle auch anhand des Namens suchen. Dies ist hilfreich, wenn Sie über eine große Anzahl von Live-Quellen verfügen.
 - Wenn Sie „Nach Namen suchen“ wählen, suchen Sie anhand des Namens nach Ihrer Live-Quelle.

5. Wählen Sie Weiter aus, um zur Registerkarte „Konfiguration planen“ zu wechseln.

6. Definieren Sie unter Wiedergabekonfiguration, wann ein Programm im Zeitplan Ihres Kanals abgespielt wird:

- Dauer in Millisekunden: Definiert die Dauer des Programms in Millisekunden. Diese Option ist nur für Programme verfügbar, die Live-Quellen verwenden.
- Übergangstyp: Definiert die Übergänge von Programm zu Programm im Zeitplan:
 - Relativ: Das Programm wird entweder vor oder nach einem anderen Programm im Zeitplan abgespielt. Diese Option ist nur für Programme verfügbar, die VOD-Quellen verwenden.
 - Absolut: Das Programm wird zu einer bestimmten Uhrzeit abgespielt. MediaTailor bemüht sich nach besten Kräften, das Programm zu der von Ihnen angegebenen Uhrzeit abzuspielen. MediaTailor startet die Wiedergabe des Programms an einer gemeinsamen Segmentgrenze zwischen dem vorherigen Programm oder Slate. Diese Option ist nur für Kanäle verfügbar, die für die Verwendung des linearen Modus konfiguriert sind [Wiedergabemodus: Der Wiedergabemodus legt das Wiedergabeverhalten des Kanals fest. MediaTailor unterstützt die folgenden Wiedergabemodi:](#).
- Startzeit des Programms: Bei absoluten Übergangstypen ist dies die Uhrzeit, zu der das Programm während der Uhrzeit, zu der die Wiedergabe des Programms geplant ist. Wenn Sie dieses Programm zu einem laufenden linearen Kanal hinzufügen, müssen Sie eine Startzeit eingeben, die 15 Minuten oder später als die aktuelle Uhrzeit liegt.
- Relative Position: Wählen Sie aus, wo das Programm relativ zu einem anderen Programm in den Zeitplan eingefügt werden soll. Sie können „Vor dem Programm“ oder „Nach

dem Programm“ wählen. Diese Einstellung gilt nicht, wenn dies das erste Programm im Programm deines Kanals ist.

- Wenn Sie „Bestehendes Programm auswählen“ wählen, wählen Sie im Menü „Bestehendes Programm verwenden“ den Programmnamen aus einer vordefinierten Liste der nächsten 100 Programme aus, die vom Kanal wiedergegeben werden.
- Wenn du „Nach einem Programm anhand des Namens suchen“ auswählst, gib den Namen eines bestehenden Programms in deinem Kanal ein.

Wenn du Werbeunterbrechungen zu deinem Programm hinzufügen möchtest, fahre mit dem nächsten Schritt fort. Werbeunterbrechungen können nur für Programme konfiguriert werden, die VOD-Quellen verwenden. Bei Live-Quellen werden Werbeunterbrechungen in DASH-Manifesten und Werbeunterbrechungen in HLS-Manifesten, die das EXT-X-DATERANGE Tag verwenden, automatisch weitergegeben.

7. Wählen Sie Weiter aus, um zu Werbeunterbrechungen hinzuzufügen zu gelangen.

8. Wählen Sie Werbeunterbrechung hinzufügen aus. Konfigurieren Sie unter Werbeunterbrechungen die Einstellungen für die Werbeunterbrechung:

- Name des Speicherorts der Slate-Quelle: Wählen Sie einen vorhandenen Quellspeicherort auswählen und wählen Sie den Quellspeicherort aus, an dem Ihr Slate gespeichert ist, den Sie zuvor in dieser Aufgabe erstellt haben.
- Name der VOD-Quelle: Wählen Sie „Eine vorhandene VOD-Quelle auswählen“ und wählen Sie die VOD-Quelle aus, die Sie für Slate verwenden und die Sie zuvor in dieser Aufgabe hinzugefügt haben. Die Dauer des Slate bestimmt die Dauer der Werbepause.
- Offset in Millisekunden: Dieser Wert bestimmt die Startzeit der Werbepause in Millisekunden als Offset im Verhältnis zum Beginn des Programms. Geben Sie einen Wert ein, der kürzer als die Dauer der VOD-Quelle ist und der an einer Segmentgrenze für alle Titel innerhalb der VOD-Quelle des Programms (alle Audio-, Video- und Untertitelspuren) ausgerichtet ist. Andernfalls wird die Werbeunterbrechung übersprungen. Wenn Sie beispielsweise 0 eingeben, entsteht eine Pre-Roll-Werbeunterbrechung, die vor Programmbeginn abgespielt wird.
- Verfügbare Zahl: MediaTailor [schreibt, in welche dieser Wert geschrieben wird](#) `splice_insert.avail_num`, wie in Abschnitt 9.7.3.1 der SCTE-35-Spezifikation, [Digital Program Insertion Cueing Message](#), definiert. Der Standardwert lautet 0. Die Werte müssen zwischen 0 und 256 (einschließlich) liegen.
- Erwartete Verfügbarkeit: MediaTailor schreibt diesen Wert `splice_insert.avails_expected`, wie in Abschnitt 9.7.3.1 der SCTE-35-Spezifikation

definiert. Der Standardwert lautet 0. Die Werte müssen zwischen 0 und 256 (einschließlich) liegen.

- ID des Splice-Ereignisses: MediaTailor schreibt diesen Wert `insplice_insert.splice_event_id`, wie in Abschnitt 9.7.3.1 der SCTE-35-Spezifikation definiert. Der Standardwert lautet 1.
- Eindeutige Programm-ID: MediaTailor schreibt diesen Wert `insplice_insert.unique_program_id`, wie in Abschnitt 9.7.3.1 der SCTE-35-Spezifikation definiert. Der Standardwert lautet 0. Die Werte müssen zwischen 0 und 256 (einschließlich) liegen.

9. Wählen Sie für einen linearen Standardkanal die Option Weiter aus, um zu Alternative Medien einstellen zu wechseln.

Weitere Informationen zur Verwendung MediaTailor zum Erstellen alternativer Medien finden Sie unter [Alternative Medien erstellen](#).

Weiterführende Informationen MediaTailor zur Personalisierung Ihrer Werbeunterbrechungen finden Sie unter [Fügen Sie personalisierte Anzeigen und Werbeunterbrechungen in einen Channel-Stream ein](#).

10. Wählen Sie Weiter aus, um zu Überprüfen und erstellen zu wechseln.
11. Wählen Sie Programm hinzufügen aus.

Weiterführende Informationen MediaTailor zur Personalisierung Ihrer Werbeunterbrechungen finden Sie unter [Fügen Sie personalisierte Anzeigen und Werbeunterbrechungen in einen Channel-Stream ein](#).

12.

 **Important**

Wenn Sie bei Kanälen, die sich in einer Endlosschleife befinden, die Programmliste für eine Sendung ändern, die innerhalb der nächsten 10 Minuten geplant ist, wird die Änderung erst in der nächsten Endlosschleife sichtbar.

Geben Sie unter Programmdetails Details zu Ihrem Programm ein:

- Name: Dies ist der Name des Programms, das du deinem Kanal hinzufügst.
- Quelltyp: Bestimmt, welche Art von Quelle das Programm wiedergibt. Diese Option ist nur für Standardkanäle verfügbar.

- VOD — Das Programm spielt eine VOD-Quelle ab, z. B. eine zuvor aufgezeichnete TV-Episode.
- Live — Das Programm spielt eine Live-Quelle ab, z. B. eine Live-Nachrichtenübertragung.
- Name des Quellverzeichnisses: Der Quellspeicherort, der dem Programm zugeordnet werden soll.

Wenn Sie „Vorhandenen Quellspeicherort auswählen“ wählen, wählen Sie im Dropdownmenü „Quellspeicherort auswählen“ einen Quellspeicherort aus. Sie können Ihren Quellstandort auch anhand des Namens suchen. Dies ist hilfreich, wenn Sie über eine große Anzahl von Quellstandorten verfügen.

Wenn Sie Geben Sie den Quellspeicherort ein wählen, suchen Sie anhand des Namens nach Ihrem Quellspeicherort.

- Name der VOD-Quelle: Der Name der VOD-Quelle, die dem Programm zugeordnet werden soll.

Wenn Sie Select a existing VOD-Quelle wählen, wählen Sie einen VOD-Quellennamen aus der Liste der VOD-Quellen, die Ihrem Konto zugeordnet sind. Sie können Ihre VOD-Quelle auch anhand des Namens suchen. Dies ist hilfreich, wenn Sie über eine große Anzahl von VOD-Quellen verfügen.

Wenn Sie „Nach Namen suchen“ wählen, suchen Sie anhand des Namens nach Ihrer VOD-Quelle.

- Name der Live-Quelle: Der Name der Live-Quelle, die dem Programm zugeordnet werden soll. Diese Option ist nur verfügbar, wenn Sie Live als Quelltyp ausgewählt haben.

Wenn Sie „Vorhandene Live-Quelle auswählen“ wählen, wählen Sie einen Namen für die Live-Quelle aus der Liste der Live-Quellen aus, die mit Ihrem Konto verknüpft sind. Sie können Ihre Live-Quelle auch anhand des Namens suchen. Dies ist hilfreich, wenn Sie über eine große Anzahl von Live-Quellen verfügen.

Wenn Sie „Nach Namen suchen“ wählen, suchen Sie anhand des Namens nach Ihrer Live-Quelle.

13. Lege unter Wiedergabekonfiguration fest, wann ein Programm im Zeitplan deines Kanals abgespielt wird:

- Dauer in Millisekunden: Definiert die Dauer des Programms in Millisekunden. Diese Option ist nur für Programme verfügbar, die Live-Quellen verwenden.

- **Übergangstyp:** Definiert die Übergänge von Programm zu Programm im Zeitplan.
 - **Relativ** — Das Programm wird entweder vor oder nach einem anderen Programm im Zeitplan abgespielt. Diese Option ist nur für Programme verfügbar, die VOD-Quellen verwenden.
 - **Absolut** — Das Programm wird zu einer bestimmten Uhrzeit abgespielt. MediaTailor bemüht sich nach besten Kräften, das Programm zu der von Ihnen angegebenen Uhrzeit abzuspielen. Wir starten die Wiedergabe des Programms an einer gemeinsamen Segmentgrenze zwischen dem vorherigen Programm oder Slate. Diese Option ist nur für Kanäle verfügbar, die für die Verwendung von konfiguriert sind [linear playback mode](#).

 Note

Beachten Sie das folgende Verhalten bei absoluten Übergangstypen:

- Wenn das vorherige Programm im Zeitplan eine Dauer hat, die über die Wanduhrzeit hinausgeht, wird das vorherige MediaTailor Programm an der gemeinsamen Segmentgrenze gekürzt, die der Wanduhr am nächsten ist.
 - Wenn zwischen den Programmen im Zeitplan Lücken bestehen, MediaTailor wird abgespielt. [filler slate](#) Wenn die Dauer der Slate kürzer als die Dauer der Lücke ist, wird die Slate in einer MediaTailor Schleife wiederholt.
- **Startzeit des Programms** — Bei absoluten Übergangstypen die Uhrzeit, zu der das Programm auf der Wanduhr abgespielt werden soll. Wenn Sie dieses Programm zu einem laufenden linearen Kanal hinzufügen, müssen Sie eine Startzeit eingeben, die 15 Minuten oder später als die aktuelle Uhrzeit liegt.
 - **Relative Position:** Wählen Sie aus, wo das Programm relativ zu einem anderen Programm in den Zeitplan eingefügt werden soll. Sie können „Vor dem Programm“ oder „Nach dem Programm“ wählen. Diese Einstellung gilt nicht, wenn dies das erste Programm im Programm Ihres Kanals ist.
 - **Relatives Programm:** Der Name des Programms, das verwendet werden soll, um das neue Programm davor oder danach einzufügen. Diese Einstellung gilt nicht, wenn dies das erste Programm im Zeitplan Ihres Kanals ist.

Wenn Sie „Bestehendes Programm auswählen“ wählen, wählen Sie im Dropdownmenü „Bestehendes Programm verwenden“ den Namen des Programms aus einer vordefinierten Liste der nächsten 100 Programme aus, die vom Kanal wiedergegeben werden.

Wenn du „Nach einem Programm anhand des Namens suchen“ auswählst, gib den Namen eines bestehenden Programms in deinem Kanal ein.

Wenn du Werbeunterbrechungen zu deinem Programm hinzufügen möchtest, fahre mit dem nächsten Schritt fort. Werbeunterbrechungen können nur für Programme konfiguriert werden, die VOD-Quellen verwenden. Bei Live-Quellen werden Werbeunterbrechungen in DASH-Manifesten und Werbeunterbrechungen in HLS-Manifesten, die das EXT-X-DATERANGE Tag verwenden, automatisch weitergegeben.

14. Wählen Sie Werbeunterbrechung hinzufügen aus. Konfigurieren Sie unter Werbeunterbrechungen die Einstellungen für die Werbeunterbrechung:

- Name des Speicherorts der Slate-Quelle: Wählen Sie einen vorhandenen Quellspeicherort auswählen und wählen Sie den Quellspeicherort aus, an dem Ihr Slate gespeichert ist, den Sie zuvor in diesem Tutorial erstellt haben.
- Name der VOD-Quelle: Wählen Sie „Eine vorhandene VOD-Quelle auswählen“ und wählen Sie die VOD-Quelle aus, die Sie für Slate verwenden und die Sie zuvor in diesem Tutorial hinzugefügt haben. Die Dauer des Slate bestimmt die Dauer der Werbepause.
- Für Offset in Millisekunden: Dieser Wert bestimmt die Startzeit der Werbeunterbrechung in Millisekunden als Offset im Verhältnis zum Beginn des Programms. Geben Sie einen Wert ein, der kürzer als die Dauer der VOD-Quelle ist und der an einer Segmentgrenze für alle Titel innerhalb der VOD-Quelle des Programms (alle Audio-, Video- und Untertitelspuren) ausgerichtet ist. Andernfalls wird die Werbeunterbrechung übersprungen. Wenn Sie beispielsweise 0 eingeben, entsteht eine Pre-Roll-Werbeunterbrechung, die vor Programmbeginn abgespielt wird.

 Note

Wenn in Ihrer VOD-Quelle Werbemarken wie DATERANGE oder EXT-X-CUE-OUT für HLS und EventStream für DASH mit einer Dauer von Null MediaTailor erkannt werden, können Sie den Offset dieser Werbemarken aus dem Drop-down-Menü auswählen, der als Offset für die Werbeunterbrechung verwendet werden soll. Damit eine Werbemöglichkeit erkannt wird, muss sie in allen Paketkonfigurationen innerhalb einer VOD-Quelle mit demselben Offset vorhanden sein und ihre Dauer muss Null sein.

- Für die Avail-Nummer wird `splice_insert.avail_num`, wie in Abschnitt 9.7.3.1 der SCTE-35-Spezifikation definiert, geschrieben. Der Standardwert ist 0. Die Werte müssen zwischen 0 und (einschließlich) 256 liegen.

Für Avail Expected ist dies wie in Abschnitt 9.7.3.1 der SCTE-35-Spezifikation definiert `splice_insert.avails_expected`, geschrieben. Der Standardwert ist 0. Die Werte müssen zwischen 0 und (einschließlich) 256 liegen.

Für die Splice-Event-ID wird dies geschrieben `splice_insert.splice_event_id`, wie in Abschnitt 9.7.3.1 der SCTE-35-Spezifikation definiert. Der Standardwert ist 1.

Bei der eindeutigen Programm-ID wird in diese geschrieben `splice_insert.unique_program_id`, wie in Abschnitt 9.7.3.1 der SCTE-35-Spezifikation definiert. Der Standardwert ist 0. Die Werte müssen zwischen 0 und (einschließlich) 256 liegen.

15. Wählen Sie Programm hinzufügen.

Weiterführende Informationen MediaTailor zur Personalisierung Ihrer Werbeunterbrechungen finden Sie unter [Fügen Sie personalisierte Anzeigen und Werbeunterbrechungen in einen Channel-Stream ein](#).

Note

Wenn dein Kanal mindestens eine Ausgabe mit einem Enhanced SCTE35 Werbemarkup-Typ hat, kannst du Metadaten für Werbeunterbrechungen einreichen. MediaTailor schreibt die eingereichten Schlüssel-Wert-Paare in das EXT-X-ASSET Tag für deine Werbepause.

Definition von Zielgruppenkohorten und alternativen Inhalten mithilfe der Programmregeln

Mit den Programmregeln können Sie Zielgruppenkohorten für einen Kanal definieren und alternative Medien angeben, die für diese Zielgruppen abgespielt werden sollen. Sie können einer Zielgruppe für ein Programm eine oder mehrere alternative Inhaltsquellen zuordnen. Nach dem Ende des Programms werden die standardmäßigen Zielgruppeninhalte wiedergegeben, sofern Sie keine weiteren alternativen Medien angeben.

Die Programmregeln sind auf Kanälen der Stufe STANDARD mit dem Wiedergabemodus LINEAR verfügbar. MediaTailorKanäle unterstützen alternative Medien für alle VOD-Quellen und Live-Quellen.

Ein Anwendungsbeispiel finden Sie unter [Programmregeln verwenden mit AWS MediaTailor](#).

Zielgruppen definieren

Definieren Sie Zielgruppen auf einem Kanal, indem Sie bei der Konfiguration eines MediaTailor Kanals die Zielgruppe nacheinander eingeben. Sie können dies entweder über die MediaTailor Konsole oder die MediaTailor CreateChannel API tun. Jede Zielgruppe muss zwischen 1 und 32 alphanumerische Zeichen lang sein. Wenn die für die Zielgruppen angegebenen Werte ungültig sind, schlägt die Anfrage fehl.

Sie können Zielgruppen nur auf Kanälen der Stufe STANDARD im Wiedergabemodus LINEAR definieren.

Wenn Sie die Zielgruppen aktualisieren müssen, können Sie dies entweder über die MediaTailor Konsole oder die MediaTailor UpdateChannel API tun.

Wenn Sie die ProgramRules Funktion verwenden, stellen Sie sicher, dass die in CreateProgram oder die UpdateProgram Anfrage AudienceMedia definierte Zielgruppe die im Kanal definierte vorhandene Zielgruppe enthält.

Alternative Medien erstellen

In der folgenden Aufgabe wird erklärt, wie alternative Medien mithilfe der MediaTailor Konsole definiert werden. Informationen zur Definition alternativer Medien mithilfe der MediaTailor API finden Sie [CreateProgram](#) in der AWS Elemental MediaTailor API-Referenz.

So definieren Sie alternative Medien in einem neuen Programm:

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Kanalmontage > Kanäle aus.
3. Wählen Sie den Kanalnamen aus, dem Sie alternative Medien hinzufügen möchten.
4. Erstellen Sie ein Programm. Weitere Informationen finden Sie unter [Ein Programm innerhalb eines Kanalzeitplans mithilfe der Konsole erstellen MediaTailor](#).
5. Alternative Medien konfigurieren:
 - Wählen Sie im Feld Zielgruppen die Option Hinzufügen aus, um die Zielgruppe auszuwählen, für die Sie alternative Medien definieren.

- Wählen Sie im Menü „Zielgruppe“ eine für den Kanal definierte Zielgruppe aus.
- Wähle Alternative Medien hinzufügen aus, um mit der Definition alternativer Medien für das Programm zu beginnen.
- MediaTailor erstellt ein Feld für alternative Medien 1. Dies ist der erste Inhalt, der im Programm als alternative Medien MediaTailor wiedergegeben wird.
- Im Feld Alternative Medien 1:
 - Wählen Sie einen Quellspeicherort aus.
 - Wählen Sie entweder einen VOD - oder einen Live-Quellentyp aus:

Für VOD

- Wählen Sie VOD als Quelltyp aus.
- (Optional) Geben Sie einen Clip-Bereich an. Bei VOD-Quellen, einschließlich VOD-Quellen für alternative Medien, können Sie angeben, dass ein Teil einer VOD-Quelle abgespielt werden soll, indem Sie am Anfang und/oder am Ende der Quelle Ausschnitte vornehmen. Geben Sie die Anfangs- und Endversätze in Millisekunden an.
- (Optional) a\ Werbeunterbrechungen hinzufügen. Dies erfolgt auf die gleiche Weise wie beim Erstellen von Programmen. Weitere Informationen finden Sie unter [Ein Programm innerhalb eines Kanalzeitplans mithilfe der Konsole erstellen MediaTailor](#).

Für Live

- Wählen Sie Live als Quelltyp aus.
- Wählen Sie eine Live-Quelle aus.
- Geben Sie eine Startzeit in Millisekunden der Uhrzeit ein, zu der diese Live-Quelle starten soll. Die Live-Quelle wird nur innerhalb des Zeitrahmens des Standardprogramms abgespielt, für das sie definiert ist. Wenn die Startzeit vor dem Start des Standardprogramms liegt, beginnt sie erst, wenn das Standardprogramm dies tut. Wenn die Startzeit nach dem Ende des Standardprogramms liegt, MediaTailor wird die Live-Quelle nicht abgespielt.
- Geben Sie eine Dauer in Millisekunden ein. Die Dauer muss mindestens 10 Minuten lang sein.
- Zusätzliche alternative Medien können diesem Programm für das Publikum hinzugefügt werden, indem Sie erneut Alternative Medien hinzufügen auswählen. Dadurch wird ein weiteres Feld mit der Bezeichnung Alternative Medien 2 erstellt. Sie können bis zu 5 alternative Medienquellen pro Programm und Zielgruppe angeben.

- Wenn Sie mit der Definition alternativer Medien für alle gewünschten Zielgruppen fertig sind, wählen Sie Weiter und fahren Sie mit der Erstellung des Programms fort.

Weitere Informationen finden Sie unter [Ein Programm innerhalb eines Kanalzeitplans mithilfe der Konsole erstellen MediaTailor](#).

Note

Alternative Medien werden nur in dem Zeitraum abgespielt, für den sie definiert sind. Wenn der gesamte alternative Inhalt den Standardinhalt überschreitet, MediaTailor wird er gekürzt. MediaTailor spielt alternative Medien in der Reihenfolge ab, in der sie definiert sind. Die Startzeiten für Live-Alternativmedien haben immer Vorrang und zuvor geplante VOD-Quellen oder Live-Quellen werden gekürzt. Jede Zeit, die nicht mit alternativen Medien für ein Publikum gefüllt ist, wird mit der vom Kanal definierten Füllliste gefüllt.

- Um Zielgruppenmedien für andere Zielgruppen zu definieren, wählen Sie neben Zielgruppen erneut Hinzufügen aus. Wählen Sie die neu erstellte Zielgruppe aus, legen Sie die Zielgruppen-ID fest und fügen Sie alternative Medien hinzu, wie oben beschrieben. Bis zu 5 Zielgruppen können alternative Medien in einem Programm verwenden.

Generierung zielgruppenspezifischer Manifeste

Verwenden Sie den `aws.mediatailor.channel.audienceId` Abfrageparameter, um ein Manifest für eine bestimmte Zielgruppe abzurufen. Dieser Abfrageparameter kann dynamisch von Ihrem CDN angehängt oder durch einen Aufruf an Ihr Content- oder Kundenverwaltungssystem hinzugefügt werden. Sie müssen die Zuordnung einer bestimmten Playback-Sitzung zu einer externen Sitzung beibehalten. `audienceId` MediaTailor Dadurch wird anstelle des Standardinhalts ein zielgruppenspezifisches Manifest abgerufen, in dem alle alternativen Medien für diese Zielgruppe definiert sind. Es ist wichtig, dass, sobald ein Manifest für eine bestimmte Zielgruppe angefordert wird, der Player das Manifest immer mit derselben Zielgruppen-ID anfordert, da es sonst zu Wiedergabefehlern kommen kann.

Wenn eine Anfrage für eine Zielgruppe gestellt wird, die auf dem Kanal nicht existiert, wird ein 404-Fehler MediaTailor zurückgegeben.

Example Ein Manifest für eine Zielgruppe abrufen

```
https:// prefix>.channel-assembly.mediatailor.us-  
west-2.amazonaws.com/v1/channel/ExampleChannel/index_dash.mpd?  
aws.mediatailor.channel.audienceId=Seattle
```

Fügen Sie personalisierte Anzeigen und Werbeunterbrechungen in einen Channel-Stream ein

Mit kannst du lineare Streams zur Kanalmontage monetarisieren MediaTailor, indem du Werbeunterbrechungen in deine Programme einfügst, ohne den Inhalt mit SCTE-35-Markern zu konditionieren. Sie können Channel Assembly mit dem MediaTailor Ad Insertion Service oder mit jeder beliebigen serverseitigen Anzeigeneinfügung (SSAI) verwenden.

In den folgenden Themen wird gezeigt, wie Sie personalisierte Anzeigen und Werbeunterbrechungen in den linearen Stream Ihres Kanals einfügen.

Themen

- [Die Anzeigeneinfügung einrichten mit MediaTailor](#)
- [SCTE-35-Nachrichten für Werbeunterbrechungen](#)

Die Anzeigeneinfügung einrichten mit MediaTailor

Um personalisierte Werbung in den Stream deines Kanals einzufügen, ist die Endpunkt-URL deines Kanals die Inhaltsquelle dafür AWS Elemental MediaTailor. In dieser Anleitung erfahren Sie, wie Sie das Einfügen von Anzeigen einrichten MediaTailor .

Voraussetzungen

Überprüfen Sie zu Beginn, ob die folgenden Anforderungen erfüllt sind:

- Bereiten Sie Ihre HLS- und DASH-Streams für die MediaTailor Anzeigenschaltung vor.
 - Falls Sie noch keine Inhaltsstreams vorbereitet haben, finden Sie weitere Informationen [Schritt 2: Bereiten Sie einen Stream vor](#) im Thema Erste Schritte mit der MediaTailor Anzeigeneinfügung.
- Haben Sie einen Ad Decision Server (ADS).
- Konfigurieren Sie die Einstellungen für Werbeunterbrechungen im Programm. Weitere Informationen finden Sie im [Configuring ad breaks for your program](#) Verfahren.

Als bewährte Methode sollten Sie die Verwendung eines Content Delivery Network (CDN) zwischen der Kanalzusammenstellung und der MediaTailor Anzeigeneinblendung in Betracht ziehen. Der Dienst zum Einfügen von MediaTailor Anzeigen kann zusätzliche Anfragen an die Herkunft generieren. Daher empfiehlt es sich, Ihr CDN so zu konfigurieren, dass es die Manifeste aus der Channel-Assembly als Proxy weiterleitet, und dann das CDN mit dem Präfix der URLs Inhaltsquellen-URL verwenden.

Für das Einfügen von Anzeigen konfigurieren MediaTailor

Im Folgenden wird gezeigt, wie Sie die MediaTailor Konsoleneinstellungen so konfigurieren, dass Sie personalisierte Werbung in den Stream Ihres Kanals einfügen können.

Um das Einfügen MediaTailor von Anzeigen zu konfigurieren

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Konfigurationen aus.
3. Geben Sie unter Erforderliche Einstellungen die grundlegenden erforderlichen Informationen zu Ihrer Konfiguration ein:
 - Name: Der Name Ihrer Konfiguration.
 - Inhaltsquelle: Gib die Wiedergabe-URL aus der Ausgabe deines Kanals ein, abzüglich des Dateinamens und der Erweiterung. Weiterführende Informationen zur MediaTailor Konfiguration finden Sie unter [Erforderliche Einstellungen](#).
 - Ad Decision Server: Geben Sie die URL für Ihr ADS ein.
4. Sie können optional die Konfigurationsalias, die Personalisierungsdetails und die erweiterten Einstellungen konfigurieren. Informationen zu diesen Einstellungen finden Sie unter [Optionale Konfigurationseinstellungen](#)
5. Wählen Sie in der Navigationsleiste Konfiguration erstellen aus.

Nachdem Sie die MediaTailor Anzeigeneinfügung eingerichtet haben, können Sie auch Werbeunterbrechungen einrichten. Detaillierte Anweisungen finden Sie unter [Erste Schritte mit dem Einfügen MediaTailor von Anzeigen](#).

SCTE-35-Nachrichten für Werbeunterbrechungen

Mit können Sie einen Inhaltskanal erstellen MediaTailor, der auf dem Quellstandort und den VOD-Quellressourcen basiert. Anschließend können Sie für jedes Programm im Zeitplan eines Kanals eine

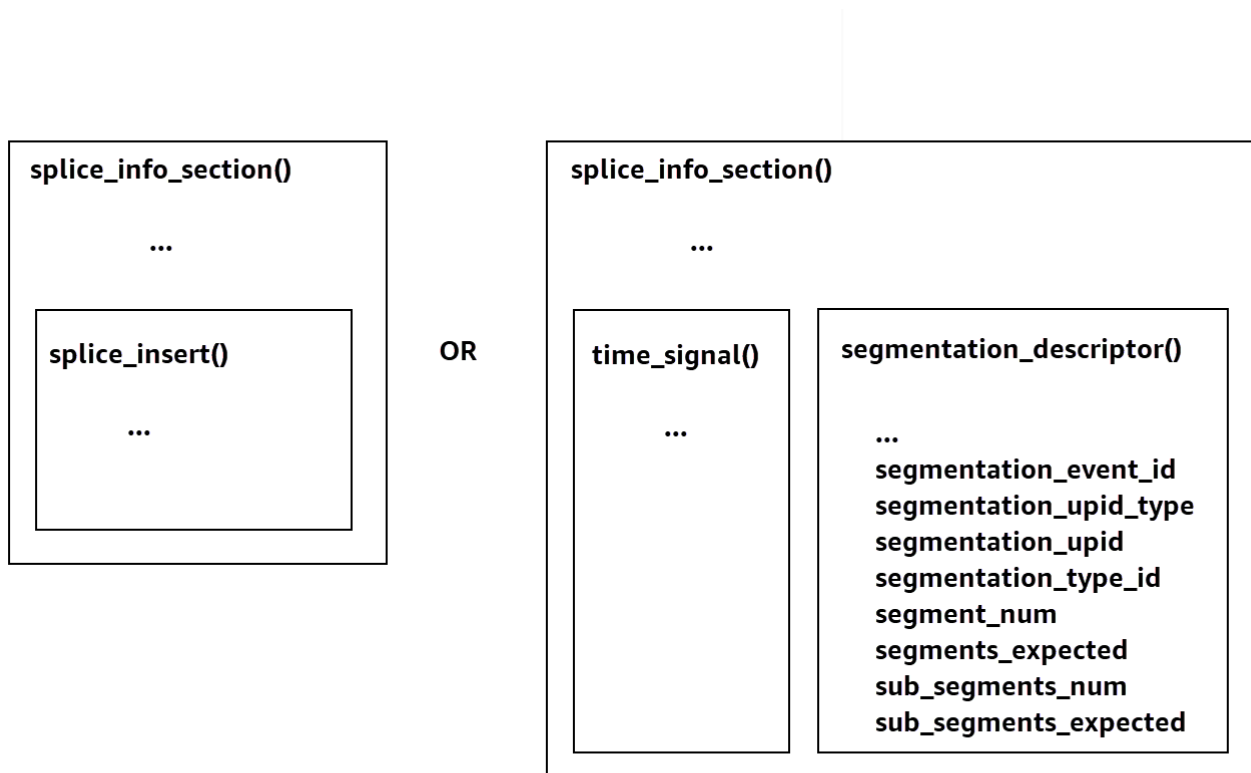
oder mehrere Werbeunterbrechungen einrichten. Sie verwenden Nachrichten, die auf der SCTE-35-Spezifikation basieren, um den Inhalt auf Werbeunterbrechungen vorzubereiten. Sie können beispielsweise SCTE-35-Nachrichten verwenden, um Metadaten zu den Werbeunterbrechungen bereitzustellen. Weitere Informationen zur SCTE-35-Spezifikation finden Sie unter [Digital Program Insertion Cueing Message](#).

Sie können die Werbeunterbrechungen auf eine von zwei Arten einrichten:

- Eine `time_signal` SCTE-35-Nachricht mit einer Nachricht anhängen.
`segmentation_descriptor` Diese `segmentation_descriptor` Nachricht enthält erweiterte Metadatenfelder wie Inhaltskennungen, die mehr Informationen über die Werbeunterbrechung enthalten. MediaTailor schreibt die Anzeigenmetadaten als Teil der SCTE-35-Daten der EXT-X-DATERANGE (HLS) oder EventStream (DASH) -Werbemarkierung in das Ausgabemanifest.
- Anhängen einer `splice_insert` SCTE-35-Nachricht, die grundlegende Metadaten zum Werbeblock enthält.
- HLS:
 - Wenn der Markup-Typ „Werbung“ lautet `Daterange`, werden Werbeunterbrechungen als EXT-X-DATERANGE Tags im MediaTailor Manifest angegeben.
 - Wenn der Markup-Typ „Werbung“ lautet `Scte35 Enhanced`, werden Werbeunterbrechungen mithilfe der folgenden Tags MediaTailor angegeben:
 - MediaTailor platziert ein Zeichen EXT-X-CUE-OUT auf dem ersten Segment des Anzeigenfensters, was auf einen Schnitt vom Inhalt bis zur Werbeunterbrechung hinweist. Es enthält die erwartete Dauer der Werbeunterbrechung, z. `EXT-X-CUE-OUT:Duration=30 B`.
 - `>EXT-X-ASSET`: Dieses Tag erscheint im selben Segment wie die Metadaten der Werbeunterbrechung, die AdBreak bei der Erstellung oder Aktualisierung des Programms angegeben wurden, EXT-X-CUE-OUT und enthält diese. Es enthält CAID immer.
 - `EXT-0ATCLS-SCTE35`: Dieses Tag erscheint im selben Segment wie die EXT-X-CUE-OUT SCTE-35-Nachricht und enthält Base64-kodierte Byte.
 - `EXT-X-CUE-OUT-CONT`: Dieses Tag erscheint in jedem nachfolgenden Segment innerhalb der Werbeanzeige und enthält Informationen zur Dauer und zur verstrichenen Zeit. Es enthält auch die Base64-kodierte SCTE-35-Nachricht und die CAID
 - `EXT-X-CUE-IN`: Dieses Tag erscheint im ersten Inhaltssegment, nachdem die Werbepause vorbei ist, und weist auf einen Schnitt von einer Werbepause zurück zum Inhalt hin.

Die folgende Abbildung zeigt die zwei Möglichkeiten, Werbeunterbrechungen in einem Kanal mithilfe von SCTE-35-Nachrichten einzurichten:

- Verwenden Sie eine `splice_insert()` Nachricht, um Werbeunterbrechungen mit grundlegenden Metadaten einzurichten.
- Verwenden Sie eine `time_signal()` Nachricht zusammen mit einer `segmentation_descriptor()` Nachricht, um Werbeunterbrechungen mit detaillierteren Metadaten einzurichten.



Informationen zur Verwendung `time_signal` finden Sie in Abschnitt 9.7.4 der SCTE-35-Spezifikation 2022, [Digital](#) Program Insertion Cueing Message.

Die Informationen zum Werbeunterbruch werden in den SCTE-35-Ausgabedaten angezeigt. `splice_info_section` Mit MediaTailor können Sie eine einzelne Nachricht mit einer einzelnen `segmentation_descriptor` `time_signal` Nachricht koppeln.

Note

Wenn Sie eine `segmentation_descriptor` Nachricht senden, müssen Sie sie als Teil des `time_signal` Nachrichtentyps senden. Die `time_signal` Nachricht enthält nur das `splice_time` Feld, das anhand eines bestimmten MediaTailor Zeitstempels erstellt wird.

In der folgenden Tabelle werden die Felder beschrieben, die für jede MediaTailor `segmentation_descriptor` Nachricht erforderlich sind. [Weitere Informationen finden Sie in Abschnitt 10.3.3.1 der SCTE-35-Spezifikation 2022, die Sie auf der ANSI Webstore-Website erwerben können.](#)

segmentation_descriptor Erforderliche Felder für eine Nachricht

Feld	Typ	Standardwert	Beschreibung
<code>segmentation_event_id</code>	Ganzzahl	1	Das ist geschrieben an <code>segmentation_descriptor.segmentation_event_id</code> .
<code>segmentation_upid_type</code>	Ganzzahl	14 (0x0E)	Das ist geschrieben. <code>segmentation_descriptor.segmentation_upid_type</code> Der Wert muss zwischen 0 und 256 (einschließlich) liegen.
<code>segmentation_upid</code>	Zeichenfolge	"" (leere Zeichenfolge)	Das ist geschrieben an <code>segmentation_descriptor.segmentation_upid</code> .

Feld	Typ	Standardwert	Beschreibung
			Der Wert muss eine hexadezimale Zeichenfolge sein, die die Zeichen 0-9 und enthält. A-F
segmentation_type_id	Ganzzahl	48 (0 x 30)	Das ist geschrieben. segmentation_descriptor.segmentation_type_id Der Wert muss zwischen 0 und 256 (einschließlich) liegen.
segment_num	Ganzzahl	0	Das ist geschrieben. segmentation_descriptor.segment_num . Der Wert muss zwischen 0 und 256 (einschließlich) liegen.
segments_expected	Ganzzahl	0	Das ist geschrieben. segmentation_descriptor.segments_expected . Der Wert muss zwischen 0 und 256 (einschließlich) liegen.

Feld	Typ	Standardwert	Beschreibung
sub_segment_num	Ganzzahl	null	Das ist geschriebensegmentation_descriptor.sub_segment_num . Der Wert muss zwischen 0 und 256 (einschließlich) liegen.
sub_segments_expected	Ganzzahl	null	Das ist geschriebensegmentation_descriptor.sub_segments_expected . Der Wert muss zwischen 0 und 256 (einschließlich) liegen.

Die folgende Tabelle zeigt die Werte, die für einige segmentation_descriptor Nachrichtfelder MediaTailor automatisch festgelegt werden.

Werte, die von MediaTailor für die Felder einer **segmentation_descriptor** Nachricht festgelegt wurden

Feld	Typ	Wert
segmentation_event_cancel_indicator	Boolesch	True
program_segmentation_flag	Boolesch	True
delivery_not_restricted_flag	Boolesch	True

MediaTailor setzt immer den Wert `segmentation_duration_flag` auf `True`. MediaTailor füllt das `segmentation_duration` Feld mit der Dauer des Statusinhalts in Ticks aus.

 Note

Beim MediaTailor Senden der `time_signal` Nachrichten wird das `splice_command_type` Feld in der `splice_info_section` Nachricht auf 6 (0x06) gesetzt.

In der HLS-Ausgabe enthält das EXT-X-DATERANGE Ausgabe-Tag für eine AdBreak `time_signal` Nachricht ein SCTE-35 Feld, das auf die serialisierte Version der Nachricht festgelegt ist.

`splice_info_section` Das folgende EXT-X-DATERANGE Tag zeigt beispielsweise die serialisierte Version der Nachricht: `splice_info_section`

```
#EXT-X-DATERANGE:ID="\1",START-DATE="\2020-09-25T02:13:20Z",DURATION=300.0,SCTE35-  
OUT=0xFC002C0000000000000000FF00506800000000001602144355454900000000100E000019BFCC00E00300000000
```

In der DASH-Ausgabe enthält das EventStream Ausgabeelement für eine AdBreak `time_signal` Nachricht ein `scte35:SpliceInfoSection` Element mit `scte35:TimeSignal` und `scte35:SegmentationDescriptor` Elementen als untergeordneten Elementen. Das `scte35:TimeSignal` Element hat ein `scte35:SpliceTime` untergeordnetes Element, und das `scte35:SegmentationDescriptor` Element hat ein untergeordnetes `scte35:SegmentationUpid` Element. Die folgende DASH-Ausgabe zeigt beispielsweise die Struktur des untergeordneten EventStream Elements des Elements:

```
<EventStream schemeIdUri="urn:scte:scte35:2013:xml" timescale="900000">
  <Event duration="270000000">
    <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="0" tier="4095">
      <scte35:TimeSignal>
        <scte35:SpliceTime ptsTime="0" />
      </scte35:TimeSignal>
      <scte35:SegmentationDescriptor segmentNum="0"
segmentationDuration="270000000" segmentationEventCancelIndicator="false"
segmentationEventId="1" segmentationTypeId="48" segmentsExpected="0">
        <scte35:SegmentationUpid segmentationUpidFormat="hexBinary"
segmentationUpidType="14">012345</scte35:SegmentationUpid>
      </scte35:SegmentationDescriptor>
    </scte35:SpliceInfoSection>
  </Event>
</EventStream>
```

Sie haben mehr über die Verwendung von SCTE-35-Nachrichten zur Einrichtung von Werbeunterbrechungen bei der Kanalzusammenstellung gelernt, die Struktur und die erforderlichen Felder für diese Nachrichten sowie Beispiele für HLS- und DASH-Ausgaben, die die SCTE-35-Nachrichten enthalten, gelernt.

Zeitversetzte Anzeige aktivieren

Zeitversetztes Ansehen bedeutet, dass Zuschauer einen Livestream zu einem früheren Zeitpunkt als dem Zeitpunkt der Anfrage ansehen können, sodass sie von Anfang an an einer Sendung teilnehmen können, die bereits läuft, oder sich eine Sendung ansehen, die bereits abgeschlossen ist. MediaTailor Kanäle unterstützen die zeitversetzte Wiedergabe von Inhalten, die bis zu 6 Stunden alt sind. Du kannst die zeitversetzte Anzeige einiger oder aller Inhalte aktivieren, indem du die maximale Zeitverzögerung für den Kanal festlegst. Clients können das Manifestfenster bis zur konfigurierten maximalen Zeitverzögerung in die Vergangenheit verschieben, indem sie gültige zeitversetzende Parameter in Anfragen angeben.

Die zeitversetzte Anzeige ist auf Kanälen der Stufe STANDARD im Wiedergabemodus LINEAR verfügbar. MediaTailor Kanäle unterstützen Timeshifting für alle VOD-Quellen und für Live-Quellen, die V1- oder MediaPackage MediaPackage V2-Ursprünge verwenden, die mit ausreichenden Startfenstern konfiguriert sind.

Important

MediaPackage V1- und MediaPackage V2-Kanäle, die als Live-Ursprünge für MediaTailor Kanäle verwendet werden, müssen über ausreichend Startfenster verfügen. Wir empfehlen ein Startfenster, das mindestens 10 Minuten länger ist als die Summe aus der maximalen Zeitverzögerung und der Dauer des Manifestfensters auf deinem Kanal. MediaTailor Informationen zur Konfiguration eines Startover-Fensters auf einem MediaPackage V1-Kanal finden Sie unter [Referenz zur zeitversetzten Anzeige](#) im V1-Benutzerhandbuch. AWS Elemental MediaPackage MediaPackage Informationen zur Konfiguration eines Startfensters auf einem MediaPackage V2-Kanal finden Sie unter [Referenz zur zeitversetzten Anzeige im AWS Elemental MediaPackage](#) V2-Benutzerhandbuch. MediaPackage

So aktivieren Sie die zeitverschobene Wiedergabe

1. Aktivieren Sie die zeitversetzte Anzeige, indem Sie einen Wert für Maximale Zeitverzögerung bei der Konfiguration eines Kanals eingeben. MediaTailor Sie können dies entweder über

die MediaTailor Konsole oder die MediaTailor API tun. Die minimal zulässige maximale Zeitverzögerung beträgt 0 Sekunden und die maximal zulässige maximale Zeitverzögerung beträgt 21600 Sekunden (6 Stunden).

Wenn Anfragen für diesen Kanal mit Zeitverschiebungsparametern MediaTailor empfängt, die der konfigurierten maximalen Zeitverzögerung entsprechen, MediaTailor wird ab dem angegebenen Zeitpunkt ein Manifest mit gleitendem Fenster generiert. Wenn die für die Time-Shifting-Parameter bereitgestellten Werte eine Verzögerung erfordern, die die maximale Zeitverzögerung überschreitet, schlagen die Anfragen fehl. Wenn die Anforderung keine zeitversetzenden Parameter enthält, generiert der Dienst ohne Verzögerung ein Manifest.

2. Stellen Sie sicher, dass Inhaltsanfragen nach Bedarf zeitversetzende Parameter enthalten. Eine Anfrage kann bis zu einen zeitversetzenden Parameter haben. Informationen zu bestimmten Zeitverschiebungsparametern finden Sie unter [Zeitversetzende Parameter für Manifestanfragen](#).

Themen

- [Zeitversetzende Parameter für Manifestanfragen](#)
- [Verwenden Sie zeitversetzte Anzeige mit CDNs](#)

Zeitversetzende Parameter für Manifestanfragen

In diesem Abschnitt sind die Parameter für zeitversetzende Manifestanfragen aufgeführt.

Um diese Funktion zu verwenden, folgen Sie den Schritten unter [Zeitversetzte Anzeige aktivieren](#).

Zeitverzögerung

Sie können eine Dauer in Sekunden angeben MediaTailor , um die Verfügbarkeit von Inhalten für Spieler zu verzögern. Das Minimum ist 0 Sekunden, und das Maximum ist die maximale Zeitverzögerung, die Sie für den Kanal konfiguriert haben.

Verwenden Sie den `aws.mediatailor.channel.timeDelay` Parameter, um den Live-Point neu zu definieren und Inhalte später verfügbar zu machen, als sie im Zeitplan Ihres Kanals erscheinen. Bei einer Zeitverzögerung von 60 Sekunden sind Inhalte, die um 12:20 Uhr im MediaTailor Zeitplan erscheinen, erst um 12:21 Uhr verfügbar. Wenn Sie Inhalte in verschiedenen Zeitzonen bereitstellen, können Sie ebenfalls eine Zeitverzögerung festlegen, die der Differenz entspricht, sodass Inhalte beispielsweise um 8:00 Uhr Ortszeit verfügbar sind.

Wenn Sie eine Zeitverzögerung für eine Manifestanforderung angeben möchten, verwenden Sie `aws.mediatailor.channel.timeDelay` Include als Abfrageparameter.

Beispiel für eine Zeitverzögerung

```
https://<some prefix>.channel-assembly.mediatailor.us-west-2.amazonaws.com/v1/channel/  
ExampleChannel/index_dash.mpd?aws.mediatailor.channel.timeDelay=901
```

Startzeit

Mit dem `aws.mediatailor.channel.startTime` Parameter können Sie einen Zeitstempel angeben, ab dem die Wiedergabe gestartet werden soll. Die Startzeit muss in einem der folgenden Formate angegeben werden:

- ISO-8601-Daten, wie `2017-08-18T21:18:54+00:00`

Alle +-Symbole in ISO-8601-Datumsangaben müssen URL-kodiert sein, wie zum Beispiel `%2B2017-08-18T21:18:54%2B00:00`

- POSIX- (oder Epoch-) Zeit, wie `1503091134`

Wenn eine Startzeit angegeben wird, MediaTailor antwortet er mit einem Manifest im Sliding-Window, als ob der Spieler das erste Manifest zur angegebenen Startzeit angefordert hätte. Beispiel: Ein Zuschauer, der mit der Wiedergabe eines Kanals beginnt `2023-10-25T14:00:00` und ein `aws.mediatailor.channel.startTime` von bereitstellt, `2023-10-25T12:00:00` sieht denselben Inhalt wie ein Zuschauer, der denselben Kanal zu schauen beginnt, ohne dass eine `2023-10-25T12:00:00` Startzeit angegeben ist. Bei Kanälen mit dem LINEAR-Wiedergabemodus ist das letzte Segment im Manifestfenster das Segment, das sich mit der Zeit überschneidet, die 10 Sekunden vor dem Zeitpunkt der Anforderung zurückliegt. Außerdem halten die Spieler einen Puffer zwischen dem Wiedergabepunkt und dem Ende des Manifestfensters bereit. Daher beginnt die Wiedergabe nicht exakt mit dem Inhalt, der für die angegebene Startzeit geplant ist.

Die Verzögerung, die sich aus der angegebenen Startzeit ergibt, muss mindestens 0 betragen und darf nicht größer sein als die maximale Zeitverzögerung, die Sie für den Kanal konfiguriert haben.

Beispiel für eine Startzeit

```
https://<some prefix>.channel-assembly.mediatailor.us-west-2.amazonaws.com/v1/channel/  
ExampleChannel/  
index_dash.mpd?aws.mediatailor.channel.startTime=2017-12-19T13:00:28-08:00
```

Programm starten

Mit dem `aws.mediatailor.channel.startProgram` Parameter können Sie ein Programm angeben, von dem aus die Wiedergabe beginnen soll. Zulässige Werte sind die Namen von Programmen, deren Startzeiten innerhalb der maximalen Zeitverzögerung liegen, die Sie für den Kanal ab dem Ende des Manifestfensters konfiguriert haben.

Wenn es mit einem Startprogramm ausgestattet ist, MediaTailor wird den Spielern der genaue Zeitpunkt angezeigt, an dem die Wiedergabe beginnen soll.

- MediaTailor Wählt für HLS-Ausgaben eine Verzögerung so, dass das erste Segment des Startprogramms 29 Sekunden vom Ende des Manifestfensters entfernt ist, und verwendet ein EXT-X-START Tag im primären Manifest, um anzugeben, dass der Player die Wiedergabe mit dem Segment beginnen soll, das 29 Sekunden vom Ende des Manifestfensters entfernt ist.
- MediaTailor Wählt für DASH-Ausgaben eine Verzögerung aus, die auf der vorgeschlagenen Präsentationsverzögerung basiert, die Sie für Ihre Ausgabe konfiguriert haben. Wenn Ihre Ausgabe eine vorgeschlagene Präsentationsverzögerung ungleich Null hat, MediaTailor wählen Sie eine Verzögerung, bei der die Dauer zwischen dem Start des ersten Segments des Startprogramms und der Uhrzeit, zu der die Anforderung MediaTailor empfangen wird, der von der Ausgabe vorgeschlagenen Präsentationsverzögerung entspricht. Andernfalls wird MediaTailor eine Verzögerung gewählt, bei der das erste Segment des Startprogramms 29 Sekunden vom Ende des Manifestfensters entfernt ist. Um optimale Ergebnisse zu erzielen, empfehlen wir Ihnen, Ihre Ausgabe so zu konfigurieren, dass eine vorgeschlagene Präsentationsverzögerung angezeigt wird, die mindestens dem Dreifachen der maximalen Segmentdauer Ihres Startprogramms plus 10 Sekunden entspricht.

Spieler halten sich nicht unbedingt an MediaTailor die Vorschläge, und der Zeitpunkt, an dem die Wiedergabe beginnt, kann leicht variieren, je nachdem, welchen Player Sie verwenden und wie Sie ihn konfigurieren. Wir empfehlen Ihnen, Ihren Kanal mit einem Startprogrammparameter in Ihrem Player zu testen und gegebenenfalls die Konfiguration Ihres Players so anzupassen, dass die Wiedergabe beim ersten Abschnitt des Startprogramms gestartet wird.

Beispiel für ein Startprogramm

```
https://<ome prefix>.channel-assembly.mediatailor.us-west-2.amazonaws.com/v1/channel/  
ExampleChannel/index_dash.mpd?aws.mediatailor.channel.startProgram=SuperBowllVII
```

Verwenden Sie zeitversetzte Anzeige mit CDNs

Um bei Angabe einer Startzeit oder eines Startprogramms ein verschiebbares Fenster zu erzielen, MediaTailor wird die Startzeit oder der Wert für das Startprogramm in eine entsprechende Zeitverzögerung umgewandelt. Der Wert dieser Zeitverzögerung hängt von der Uhrzeit ab, zu der der Player oder das CDN das Manifest anfordert. Aus diesem Grund müssen Sie bei der Verwendung eines CDN mit MediaTailor den Parametern Startzeit oder Startprogramm das entsprechende Caching-Verhalten auf Ihrem CDN konfigurieren.

Um diese Funktion zu verwenden, folgen Sie den Schritten unter. [Zeitversetzte Anzeige aktivieren](#)

HLS-Beispiel

Angenommen, Sie fordern mithilfe einer URL wie der folgenden ein primäres HLS-Manifest mit einer Startzeit an:

```
https://<some prefix>.channel-assembly.mediatailor.us-west-2.amazonaws.com/v1/channel/  
ExampleChannel/  
index_hls.m3u8?aws.mediatailor.channel.startTime=2017-12-19T13:00:28-08:00
```

MediaTailor antwortet mit einem Manifest, das Zeitverzögerungsparameter für das untergeordnete Manifest enthält. URLs Wenn Sie das Manifest beispielsweise zu einem Zeitpunkt anfordern 2017-12-19T13:20:28-08:00, also 1200 Sekunden nach der angeforderten Startzeit, MediaTailor antworten Sie mit einem primären Manifest wie dem folgenden:

```
#EXTM3U  
#EXT-X-VERSION:6  
#EXT-X-STREAM-INF:CODECS="avc1.4D401F,mp4a.40.2",AVERAGE-  
BANDWIDTH=1426714,RESOLUTION=852x480,FRAME-RATE=30.0,BANDWIDTH=1493368  
index_hls/1.m3u8?aws.mediatailor.channel.timeDelay=1200  
#EXT-X-STREAM-INF:CODECS="avc1.4D401E,mp4a.40.2",AVERAGE-  
BANDWIDTH=986714,RESOLUTION=640x360,FRAME-RATE=30.0,BANDWIDTH=1024034  
index_hls/2.m3u8?aws.mediatailor.channel.timeDelay=1200  
#EXT-X-STREAM-INF:CODECS="avc1.4D400D,mp4a.40.2",AVERAGE-  
BANDWIDTH=476305,RESOLUTION=320x240,FRAME-RATE=30.0,BANDWIDTH=498374  
index_hls/3.m3u8?aws.mediatailor.channel.timeDelay=1200
```

DASH-Beispiel

Angenommen, Sie fordern mit einer URL wie der folgenden ein DASH-Manifest mit einer Startzeit an:

```
https://<some prefix>.channel-assembly.mediatailor.us-west-2.amazonaws.com/v1/channel/  
ExampleChannel/  
index_dash.mpd?aws.mediatailor.channel.startTime=2017-12-19T13:00:28-08:00
```

MediaTailor antwortet mit einer Weiterleitung auf dasselbe Manifest, jedoch mit einer Zeitverzögerung statt einer Startzeit. Wenn Sie das Manifest beispielsweise zu einem Zeitpunkt anfordern 2017-12-19T13:20:28-08:00, der 1200 Sekunden nach der angeforderten Startzeit liegt, dann MediaTailor antwortet es mit dem HTTP-Status 302 Found und einem Location Header mit Wert `./index_dash.mpd?aws.mediatailor.channel.timeDelay=1200`.

Anforderungen an die CDN-Konfiguration

Wenn Sie zeitversetzende Abfrageparameter mit einem CDN verwenden, empfehlen wir, Ihr CDN wie folgt zu konfigurieren:

- Wenn Sie zeitversetzende Abfrageparameter verwenden, nehmen Sie diese Parameter in den Cache-Schlüssel Ihres CDN auf. Nehmen Sie außerdem den Abfrageparameter für die Zeitverzögerung in den Cache-Schlüssel Ihres CDN auf, wenn Sie zeitversetzende Parameter verwenden.
- Wenn Sie einen der Abfrageparameter Startzeit oder Startprogramm verwenden, gilt Folgendes:
 - Für HLS konfigurieren Sie Ihr CDN so, dass primäre Manifeste nicht länger als eine typische Segmentdauer auf Ihrem Kanal zwischengespeichert werden.
 - Für DASH konfigurieren Sie Ihr CDN so, dass Weiterleitungen mit dem HTTP-Status 302 nicht länger als eine typische Segmentdauer auf Ihrem Kanal zwischengespeichert werden und dass solche Weiterleitungen an den Player weitergeleitet werden.

Informationen zur Konfiguration von Caching bei Amazon CloudFront finden Sie unter [Managing how long content stay in the cache \(Ablauf\)](#) im CloudFront Entwicklerhandbuch. Informationen darüber, wie Amazon CloudFront mit Weiterleitungen umgeht, finden Sie unter [So CloudFront verarbeitet Amazon HTTP 3xx-Statuscodes von Ihrem Absender](#) im CloudFront Entwicklerhandbuch.

Behebung von Wiedergabefehlern, die zurückgegeben wurden von MediaTailor

Dieser Abschnitt enthält Informationen über die HTTP-Fehlercodes, die Sie möglicherweise beim Testen Ihrer Player-Software und während der normalen Verarbeitung von Player-Anforderungen erhalten.

Note

Möglicherweise erhalten Sie auch Fehler von der AWS Elemental MediaTailor API bei Konfigurationsvorgängen wie `PutPlaybackConfiguration` und `GetPlaybackConfiguration`. Informationen zu diesen Fehlertypen finden Sie in der [AWS Elemental MediaTailor API-Referenz](#).

Wenn dein Spieler direkt oder über ein CDN eine Anfrage an sendet, MediaTailor antwortet er mit einem Statuscode. AWS Elemental MediaTailor Wenn die Anfrage MediaTailor erfolgreich bearbeitet wurde, wird der HTTP-Statuscode 200 OK, der den Erfolg anzeigt, zusammen mit dem ausgefüllten Manifest zurückgegeben. Wenn die Anfrage nicht erfolgreich ist, werden ein HTTP-Statuscode, ein Ausnahmename und eine Fehlermeldung MediaTailor zurückgegeben.

AWS Elemental MediaTailor gibt zwei Klassen von Fehlern zurück:

- **Client-Fehler** — Fehler, die normalerweise durch ein Problem in der Anfrage selbst verursacht werden, z. B. durch eine falsch formatierte Anfrage, einen ungültigen Parameter oder eine fehlerhafte URL. Diese Fehler haben einen HTTP- 4xx-Antwort-Code.
- **Serverfehler** — Fehler, die normalerweise durch ein Problem mit MediaTailor oder einer seiner Abhängigkeiten verursacht werden, z. B. durch den Ad Decision Server (ADS) oder den Ursprungsserver. Diese Fehler haben einen HTTP- 5xx-Antwort-Code.

Themen

- [Wiedergabefehler auf dem Client wurden zurückgegeben von AWS Elemental MediaTailor](#)
- [Fehler bei der Serverwiedergabe wurden zurückgegeben von AWS Elemental MediaTailor](#)
- [Beispiele für Fehler bei der Wiedergabe](#)

Wiedergabefehler auf dem Client wurden zurückgegeben von AWS Elemental MediaTailor

Allgemeine Hinweise:

- Detaillierte Informationen zu den meisten Fehlern finden Sie in den Kopfzeilen und im Hauptteil der Antwort.
- Bei einigen Fehlern müssen Sie Ihre Konfigurationseinstellungen überprüfen. Sie können die Einstellungen für Ihre Wiedergabekonfiguration unter abrufen AWS Elemental MediaTailor. Für die API ist die `ResourceGetPlaybackConfiguration/Name`. Einzelheiten finden Sie in der [AWS Elemental MediaTailor API-Referenz](#).

In der folgenden Tabelle sind die Client-Fehlercodes aufgeführt, die durch die Manifest-Manipulationsaktivitäten zurückgegeben werden AWS Elemental MediaTailor, mögliche Ursachen und Maßnahmen, die Sie zu ihrer Behebung ergreifen können.

Code	Name der Ausnahme	Bedeutung	Vorgehensweise
400	<code>BadRequestException</code>	MediaTailor kann die Anfrage aufgrund eines oder mehrerer Formatierungs- oder Inhaltsfehler nicht bearbeiten. Ein Parameter ist möglicherweise nicht ordnungsgemäß formatiert oder die Anforderung enthält möglicherweise eine ungültige Playback-Konfiguration oder Sitzungs-ID.	Stellen Sie sicher, dass Ihre Anforderung ordnungsgemäß formatiert ist und korrekte Informationen enthält. Stellen Sie sicher, dass die Einstellung des Wiedergabe-Endpunkts am Player mit der ManifestEndpointPrefix -Einstellung übereinstimmt, die von <code>GetPlaybackConfiguration</code> zurückgegeben wird. Wiederholen Sie Ihre Anforderung.
400	<code>AccessDeniedException</code>	Der in der Anforderung angegebene Host-Header stimmt nicht mit dem Präfix des Manifest-	Überprüfen Sie Ihre CDN-Einstellungen und stellen Sie sicher, dass Sie das richtige

Code	Name der Ausnahme	Bedeutung	Vorgehensweise
		Endpunkts über ein, das in der MediaTailor-Wiedergabe-URL konfiguriert ist. Ihr CDN ist möglicherweise falsch konfiguriert.	Präfix des Manifest-Endpunkts für MediaTailor verwenden. Wiederholen Sie Ihre Anforderung.
40	NotFoundException	MediaTailor kann die angegebenen Informationen nicht finden. Mögliche Gründe sind eine URL, die keiner Entsprechung im Service zugeordnet ist, eine nicht definierte Konfiguration oder eine nicht verfügbare Sitzung.	Überprüfen Sie Ihre Konfiguration und die Gültigkeit Ihrer Anforderung und initialisieren Sie die Session dann neu.
40	ConflictException	Ein Player hat versucht, für eine einzige Sitzung mehrere Wiedergabelisten gleichzeitig zu laden. Infolgedessen MediaTailor wurde ein Sitzungskonsistenzkonflikt festgestellt. Dieses Problem tritt bei HLS-Playern auf.	Stellen Sie sicher, dass Ihr Player nur jeweils eine Wiedergabeliste anfordert. Dies stimmt mit der HLS-Spezifikation überein.
41	Gone	Ein AWS Support-Mitarbeiter hat eine Spiellersitzung oder Kundenkonfiguration blockiert. AWS Der Support tut dies in seltenen Fällen, wenn wir eine sehr hohe Anzahl von 4xx-Anfragen feststellen, die auf fehlerhaften Datenverkehr für eine einzelne Sitzung oder Konfiguration zurückzuführen sind.	Wenn Sie der Meinung sind, dass die Anfrage nicht blockiert werden sollte, wenden Sie sich an den AWS-Support . Sie können das überprüfen und nötigenfalls den Blockfilter entfernen.

Wenn Sie weitere Support benötigen, wenden Sie sich an den [AWS-Support](#).

Fehler bei der Serverwiedergabe wurden zurückgegeben von AWS Elemental MediaTailor

Allgemeine Hinweise:

- Detaillierte Informationen zu den meisten Fehlern finden Sie in den Kopfzeilen und im Hauptteil der Antwort.
- Bei einigen Fehlern müssen Sie Ihre Konfigurationseinstellungen überprüfen. Sie können die Einstellungen für Ihre Wiedergabekonfiguration unter abrufen AWS Elemental MediaTailor. Für die API ist die `ResourceGetPlaybackConfiguration/Name`. Einzelheiten finden Sie in der [AWS Elemental MediaTailor API-Referenz](#).

In der folgenden Tabelle sind die Serverfehlercodes aufgeführt, die von den Aktivitäten zur Manipulation von Manifesten zurückgegeben wurden AWS Elemental MediaTailor, mögliche Ursachen und Maßnahmen, die Sie zu ihrer Behebung ergreifen können.

Code	Name der Ausnahme	Bedeutung	Vorgehensweise
50	InternalServerError	Unbehandelte Ausnahme	Wiederholen Sie die Anforderung. Wenn das Problem weiterhin besteht, überprüfen Sie den gemeldeten Status von MediaTailor für Ihre AWS Region unter https://status.aws.amazon.com/ .
50	BadGatewayException	Entweder die Adresse des Ursprungsservers oder die Adresse des Ad Decision Servers (ADS) ist ungültig. Beispiele für ungültige Adressen sind eine private IP-Adresse und localhost .	Stellen Sie sicher, dass Ihre Konfiguration über die richtigen Einstellungen für Ihren ADS und Ursprungs-Server verfügt, und wiederholen Sie die Anforderung.

Code	Name der Ausnahme	Bedeutung	Vorgehensweise
50	UnsupportedManifestException	Entweder wurde das Ursprungsmanifest geändert, sodass es nicht personalisiert werden MediaTailor kann, oder es unterstützt das Manifestformat des Ursprungs MediaTailor nicht.	Dies wirkt sich möglicherweise nur auf diese einzelne Sitzung aus. Initialisieren Sie die Sitzung erneut. Sie können dies gewöhnlich erreichen, indem Sie die Seite auf der Anzeige aktualisieren. Wenn das Problem weiterhin besteht, stellen Sie sicher, dass das Manifest-Format des Ursprungs-Servers von MediaTailor unterstützt wird. Weitere Informationen finden Sie unter Integration einer Inhaltsquelle .
50	LoadShed	MediaTailor bei der Bearbeitung Ihrer Anfrage kam es zu einer Ressourcenbeschränkung.	Wiederholen Sie die Anforderung. Falls das Problem weiterhin besteht, überprüfen Sie den MediaTailor für Ihre AWS Region gemeldeten Status unter https://status.aws.amazon.com/ .
50	ThrottlingException	Da Ihre Transaktionen pro Sekunde das für Sie geltende Kontingent erreicht haben, wird Ihre Nutzung von MediaTailor abgelehnt.	Wiederholen Sie die Anforderung. Sie können den gemeldeten Zustand von MediaTailor für Ihre AWS Region auch unter https://status.aws.amazon.com/ überprüfen. Möglicherweise möchten Sie das Kontingent für Ihre Transaktionen pro Sekunde erhöhen. Weitere Informationen finden Sie unter the section called "Kontingente für das Einfügen von Anzeigen" .

Code	Name der Ausnahme	Bedeutung	Vorgehensweise
50	GatewayTimeoutException	Bei der Kontaktaufnahme mit dem Ursprungsserver ist ein Timeout aufgetreten MediaTailor .	Wiederholen Sie die Anforderung. Wenn das Problem weiterhin besteht, überprüfen Sie den Zustand des Ursprungs-Servers und stellen Sie sicher, dass der Ursprungs-Server innerhalb des Timeouts des Ursprungs-Servers für Inhalte antwortet, das unter the section called “Kontingente für das Einfügen von Anzeigen” aufgeführt wird.

Wenn Sie weitere Support benötigen, wenden Sie sich an den [AWS-Support](#).

Beispiele für Fehler bei der Wiedergabe

In diesem Abschnitt sind einige Beispiele für Wiedergabefehler aufgeführt, die bei Interaktionen mit der Befehlszeile auftreten können AWS Elemental MediaTailor.

Das folgende Beispiel zeigt das Ergebnis, wenn ein Timeout zwischen dem AWS Elemental MediaTailor Ad Decision Server (ADS) oder dem Ursprungsserver auftritt.

```
~[ ]> curl -vvv https://111122223333444455556666123456789012.mediatailor.us-west-2.amazonaws.com/v1/master/123456789012/Multiperiod_DASH_Demo/index.mpd
* Trying 54.186.133.224...
* Connected to 111122223333444455556666123456789012.mediatailor.us-west-2.amazonaws.com (11.222.333.444) port 555 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
* Server certificate: mediatailor.us-west-2.amazonaws.com
* Server certificate: Amazon
* Server certificate: Amazon Root CA 1
* Server certificate: Starfield Services Root Certificate Authority - G2
> GET /v1/master/123456789012/Multiperiod_DASH_Demo/index.mpd HTTP/1.1
> Host: 111122223333444455556666123456789012.mediatailor.us-west-2.amazonaws.com
> User-Agent: curl/7.43.0
> Accept: */*
```

```
>  
< HTTP/1.1 504 Gateway Timeout  
< Date: Thu, 29 Nov 2018 18:43:14 GMT  
< Content-Type: application/json  
< Content-Length: 338  
< Connection: keep-alive  
< x-amzn-RequestId: 123456789012-123456789012  
< x-amzn-ErrorType: GatewayTimeoutException:http://internal.amazon.com/coral/  
com.amazon.elemental.midas.mms.coral/  
<  
* Connection #0 to host 111122223333444455556666123456789012.mediatailor.us-  
west-2.amazonaws.com left intact  
{"message":"failed to generate manifest: Unable to obtain template playlist.  
origin URL:[https://777788889999.mediapackage.us-west-2.amazonaws.com/out/  
v1/444455556666111122223333/index.mpd], asset path: [index.mpd], sessionId:  
[123456789012123456789012] customerId:[123456789012]}%"
```

Sicherheit in AWS Elemental MediaTailor

Cloud-Sicherheit hat AWS höchste Priorität. Als AWS Kunde profitieren Sie von Rechenzentren und Netzwerkarchitekturen, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame AWS Verantwortung von Ihnen und Ihnen. Das [Modell der geteilten Verantwortung](#) beschreibt dies als Sicherheit der Cloud selbst und Sicherheit in der Cloud:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, auf der AWS Dienste in der ausgeführt AWS Cloud werden. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Externe Prüfer testen und verifizieren regelmäßig die Wirksamkeit unserer Sicherheitsmaßnahmen im Rahmen der [AWS](#) . Weitere Informationen zu den Compliance-Programmen, die für gelten AWS Elemental MediaTailor, finden Sie unter [AWS Services im Umfang nach Compliance-Programmen AWS](#) .
- Sicherheit in der Cloud — Ihre Verantwortung richtet sich nach dem AWS Dienst, den Sie nutzen. Sie sind auch für andere Faktoren verantwortlich, etwa für die Vertraulichkeit Ihrer Daten, für die Anforderungen Ihres Unternehmens und für die geltenden Gesetze und Vorschriften.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Nutzung anwenden können MediaTailor. In den folgenden Themen erfahren Sie, wie Sie die Konfiguration vornehmen MediaTailor , um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie erfahren auch, wie Sie andere AWS Dienste nutzen können, die Sie bei der Überwachung und Sicherung Ihrer MediaTailor Ressourcen unterstützen.

Themen

- [Datenschutz in AWS Elemental MediaTailor](#)
- [Identity and Access Management für AWS Elemental MediaTailor](#)
- [Konformitätsprüfung für AWS Elemental MediaTailor](#)
- [Resilienz in AWS Elemental MediaTailor](#)
- [Sicherheit der Infrastruktur in MediaTailor](#)
- [Serviceübergreifende Confused-Deputy-Prävention](#)
- [Anmeldung und Überwachung MediaTailor](#)

Datenschutz in AWS Elemental MediaTailor

Das [Modell der AWS gemeinsamen Verantwortung](#) und geteilter Verantwortung gilt für den Datenschutz in AWS Elemental MediaTailor. Wie in diesem Modell beschrieben, AWS ist verantwortlich für den Schutz der globalen Infrastruktur, auf der alle Systeme laufen AWS Cloud. Sie sind dafür verantwortlich, die Kontrolle über Ihre in dieser Infrastruktur gehosteten Inhalte zu behalten. Sie sind auch für die Sicherheitskonfiguration und die Verwaltungsaufgaben für die von Ihnen verwendeten AWS-Services verantwortlich. Weitere Informationen zum Datenschutz finden Sie unter [Häufig gestellte Fragen zum Datenschutz](#). Informationen zum Datenschutz in Europa finden Sie im Blog-Beitrag [AWS -Modell der geteilten Verantwortung und in der DSGVO](#) im AWS - Sicherheitsblog.

Aus Datenschutzgründen empfehlen wir, dass Sie AWS-Konto Anmeldeinformationen schützen und einzelne Benutzer mit AWS IAM Identity Center oder AWS Identity and Access Management (IAM) einrichten. So erhält jeder Benutzer nur die Berechtigungen, die zum Durchführen seiner Aufgaben erforderlich sind. Außerdem empfehlen wir, die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto die Multi-Faktor-Authentifizierung (MFA).
- Wird verwendet SSL/TLS , um mit AWS Ressourcen zu kommunizieren. Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein AWS CloudTrail. Informationen zur Verwendung von CloudTrail Pfaden zur Erfassung von AWS Aktivitäten finden Sie unter [Arbeiten mit CloudTrail Pfaden](#) im AWS CloudTrail Benutzerhandbuch.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen darin enthaltenen Standardsicherheitskontrollen AWS-Services.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu schützen.
- Wenn Sie für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine API FIPS 140-3-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter [Federal Information Processing Standard \(FIPS\) 140-3](#).

Wir empfehlen dringend, in Freitextfeldern, z. B. im Feld Name, keine vertraulichen oder sensiblen Informationen wie die E-Mail-Adressen Ihrer Kunden einzugeben. Dies gilt auch, wenn Sie mit der Konsole, der API oder auf andere AWS-Services Weise arbeiten oder diese verwenden. AWS CLI AWS SDKs Alle Daten, die Sie in Tags oder Freitextfelder eingeben, die für Namen verwendet

werden, können für Abrechnungs- oder Diagnoseprotokolle verwendet werden. Wenn Sie eine URL für einen externen Server bereitstellen, empfehlen wir dringend, keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL einzuschließen.

Datenverschlüsselung

AWS Elemental MediaTailor verschlüsselt oder entschlüsselt keine Daten bei der Verwaltung von Inhaltsverzeichnissen oder bei der Kommunikation mit Servern oder CDNs Playern. MediaTailor verlangt nicht, dass Sie Kundendaten oder andere vertrauliche Informationen angeben.

Geben Sie keine vertraulichen Informationen wie Kundenkontonummern, Kreditkarteninformationen oder Anmeldeinformationen in Freiformfelder oder Abfrageparameter ein. Dies gilt für die gesamte Verwendung von AWS Elemental MediaTailor, einschließlich der Konsole, der API und der AWS Command Line Interface (AWS CLI). SDKs Alle Daten, die Sie in den Service eingeben, werden möglicherweise in Diagnoseprotokolle aufgenommen.

Wenn Sie eine URL für einen externen Server bereitzustellen, schließen Sie keine unverschlüsselten Anmeldeinformationen zur Validierung Ihrer Anforderung bei dem betreffenden Server ein.

Identity and Access Management für AWS Elemental MediaTailor

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service, den Zugriff auf AWS Ressourcen sicher zu kontrollieren. IAM-Administratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um Ressourcen zu verwenden. MediaTailor IAM ist ein Programm AWS-Service, das Sie ohne zusätzliche Kosten nutzen können.

Themen

- [Zielgruppe](#)
- [Authentifizierung mit Identitäten](#)
- [Verwalten des Zugriffs mit Richtlinien](#)
- [Wie AWS Elemental MediaTailor funktioniert mit IAM](#)
- [Beispiele für identitätsbasierte Richtlinien für AWS Elemental MediaTailor](#)
- [Beispiele für ressourcenbasierte Richtlinien für AWS Elemental MediaTailor](#)
- [AWS verwaltete Richtlinien für AWS Elemental MediaTailor](#)

- [Verwenden von serviceverknüpften Rollen für MediaTailor](#)
- [Fehlerbehebung bei AWS Elemental MediaTailor Identität und Zugriff](#)

Zielgruppe

Die Art und Weise, wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, in der Sie tätig sind. MediaTailor

Dienstbenutzer — Wenn Sie den MediaTailor Dienst für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die erforderlichen Anmeldeinformationen und Berechtigungen zur Verfügung. Wenn Sie für Ihre Arbeit mehr MediaTailor Funktionen verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Funktionsweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anfordern müssen. Unter [Fehlerbehebung bei AWS Elemental MediaTailor Identität und Zugriff](#) finden Sie nützliche Informationen für den Fall, dass Sie keinen Zugriff auf eine Feature in MediaTailor haben.

Serviceadministrator — Wenn Sie in Ihrem Unternehmen für die MediaTailor Ressourcen verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf MediaTailor. Es ist Ihre Aufgabe, zu bestimmen, auf welche MediaTailor Funktionen und Ressourcen Ihre Servicebenutzer zugreifen sollen. Anschließend müssen Sie Anforderungen an Ihren IAM-Administrator senden, um die Berechtigungen der Servicebenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die Grundkonzepte von IAM nachzuvollziehen. Weitere Informationen darüber, wie Ihr Unternehmen IAM nutzen kann MediaTailor, finden Sie unter [Wie AWS Elemental MediaTailor funktioniert mit IAM](#).

IAM-Administrator: Wenn Sie als IAM-Administrator fungieren, sollten Sie Einzelheiten dazu kennen, wie Sie Richtlinien zur Verwaltung des Zugriffs auf MediaTailor verfassen können. Beispiele für MediaTailor identitätsbasierte Richtlinien, die Sie in IAM verwenden können, finden Sie unter [Beispiele für identitätsbasierte Richtlinien für AWS Elemental MediaTailor](#)

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen als IAM-Benutzer authentifiziert (angemeldet AWS) sein oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich AWS als föderierte Identität anmelden, indem Sie Anmeldeinformationen verwenden, die über eine Identitätsquelle bereitgestellt wurden. AWS IAM Identity Center (IAM Identity Center) -Benutzer, die Single Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie

sich als Verbundidentität anmelden, hat der Administrator vorher mithilfe von IAM-Rollen einen Identitätsverbund eingerichtet. Wenn Sie über den Verbund darauf zugreifen AWS , übernehmen Sie indirekt eine Rolle.

Je nachdem, welcher Benutzertyp Sie sind, können Sie sich beim AWS Management Console oder beim AWS Zugangsportal anmelden. Weitere Informationen zur Anmeldung finden Sie AWS unter [So melden Sie sich bei Ihrem an AWS-Konto](#) im AWS-Anmeldung Benutzerhandbuch.

Wenn Sie AWS programmgesteuert darauf zugreifen, AWS stellt es ein Software Development Kit (SDK) und eine Befehlszeilenschnittstelle (CLI) bereit, um Ihre Anfragen mithilfe Ihrer Anmeldeinformationen kryptografisch zu signieren. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode für die Selbstsignierung von Anforderungen finden Sie unter [AWS Signature Version 4 für API-Anforderungen](#) im IAM-Benutzerhandbuch.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen bereitstellen. AWS empfiehlt beispielsweise, die Multi-Faktor-Authentifizierung (MFA) zu verwenden, um die Sicherheit Ihres Kontos zu erhöhen. Weitere Informationen finden Sie unter [Multi-Faktor-Authentifizierung](#) im AWS IAM Identity Center - Benutzerhandbuch und [AWS Multi-Faktor-Authentifizierung \(MFA\) in IAM](#) im IAM-Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie ein AWS-Konto erstellen, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle AWS-Services Ressourcen im Konto hat. Diese Identität wird als AWS-Konto Root-Benutzer bezeichnet. Der Zugriff erfolgt, indem Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, mit denen Sie das Konto erstellt haben. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen. Verwenden Sie diese nur, um die Aufgaben auszuführen, die nur der Root-Benutzer ausführen kann. Eine vollständige Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter [Aufgaben, die Root-Benutzer-Anmeldeinformationen erfordern](#) im IAM-Benutzerhandbuch.

Verbundidentität

Als bewährte Methode sollten menschliche Benutzer, einschließlich Benutzer, die Administratorzugriff benötigen, für den Zugriff AWS-Services mithilfe temporärer Anmeldeinformationen den Verbund mit einem Identitätsanbieter verwenden.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensbenutzerverzeichnis, einem Web-Identitätsanbieter AWS Directory Service, dem Identity Center-Verzeichnis oder einem beliebigen Benutzer, der mithilfe AWS-Services von Anmeldeinformationen zugreift, die über eine Identitätsquelle bereitgestellt wurden. Wenn föderierte Identitäten darauf zugreifen AWS-Konten, übernehmen sie Rollen, und die Rollen stellen temporäre Anmeldeinformationen bereit.

Für die zentrale Zugriffsverwaltung empfehlen wir Ihnen, AWS IAM Identity Center zu verwenden. Sie können Benutzer und Gruppen in IAM Identity Center erstellen, oder Sie können eine Verbindung zu einer Gruppe von Benutzern und Gruppen in Ihrer eigenen Identitätsquelle herstellen und diese synchronisieren, um sie in all Ihren AWS-Konten Anwendungen zu verwenden. Informationen zu IAM Identity Center finden Sie unter [Was ist IAM Identity Center?](#) im AWS IAM Identity Center - Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein [IAM-Benutzer](#) ist eine Identität innerhalb Ihres Unternehmens AWS-Konto, die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. Wenn möglich, empfehlen wir, temporäre Anmeldeinformationen zu verwenden, anstatt IAM-Benutzer zu erstellen, die langfristige Anmeldeinformationen wie Passwörter und Zugriffsschlüssel haben. Bei speziellen Anwendungsfällen, die langfristige Anmeldeinformationen mit IAM-Benutzern erfordern, empfehlen wir jedoch, die Zugriffsschlüssel zu rotieren. Weitere Informationen finden Sie unter [Regelmäßiges Rotieren von Zugriffsschlüsseln für Anwendungsfälle, die langfristige Anmeldeinformationen erfordern](#) im IAM-Benutzerhandbuch.

Eine [IAM-Gruppe](#) ist eine Identität, die eine Sammlung von IAM-Benutzern angibt. Sie können sich nicht als Gruppe anmelden. Mithilfe von Gruppen können Sie Berechtigungen für mehrere Benutzer gleichzeitig angeben. Gruppen vereinfachen die Verwaltung von Berechtigungen, wenn es zahlreiche Benutzer gibt. Sie könnten beispielsweise eine Gruppe benennen IAMAdmins und dieser Gruppe Berechtigungen zur Verwaltung von IAM-Ressourcen erteilen.

Benutzer unterscheiden sich von Rollen. Ein Benutzer ist einer einzigen Person oder Anwendung eindeutig zugeordnet. Eine Rolle kann von allen Personen angenommen werden, die sie benötigen. Benutzer besitzen dauerhafte Anmeldeinformationen. Rollen stellen temporäre Anmeldeinformationen bereit. Weitere Informationen finden Sie unter [Anwendungsfälle für IAM-Benutzer](#) im IAM-Benutzerhandbuch.

IAM-Rollen

Eine [IAM-Rolle](#) ist eine Identität innerhalb von Ihrem AWS-Konto, die über bestimmte Berechtigungen verfügt. Sie ist einem IAM-Benutzer vergleichbar, jedoch nicht mit einer bestimmten Person verknüpft. Um vorübergehend eine IAM-Rolle in der zu übernehmen AWS Management Console, können Sie [von einer Benutzer- zu einer IAM-Rolle \(Konsole\) wechseln](#). Sie können eine Rolle übernehmen, indem Sie eine AWS CLI oder AWS API-Operation aufrufen oder eine benutzerdefinierte URL verwenden. Weitere Informationen zu Methoden für die Verwendung von Rollen finden Sie unter [Methoden für die Übernahme einer Rolle](#) im IAM-Benutzerhandbuch.

IAM-Rollen mit temporären Anmeldeinformationen sind in folgenden Situationen hilfreich:

- **Verbundbenutzerzugriff** – Um einer Verbundidentität Berechtigungen zuzuweisen, erstellen Sie eine Rolle und definieren Berechtigungen für die Rolle. Wird eine Verbundidentität authentifiziert, so wird die Identität der Rolle zugeordnet und erhält die von der Rolle definierten Berechtigungen. Informationen zu Rollen für den Verbund finden Sie unter [Erstellen von Rollen für externe Identitätsanbieter \(Verbund\)](#) im IAM-Benutzerhandbuch. Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Wenn Sie steuern möchten, worauf Ihre Identitäten nach der Authentifizierung zugreifen können, korreliert IAM Identity Center den Berechtigungssatz mit einer Rolle in IAM. Informationen zu Berechtigungssätzen finden Sie unter [Berechtigungssätze](#) im AWS IAM Identity Center -Benutzerhandbuch.
- **Temporäre IAM-Benutzerberechtigungen** – Ein IAM-Benutzer oder eine -Rolle kann eine IAM-Rolle übernehmen, um vorübergehend andere Berechtigungen für eine bestimmte Aufgabe zu erhalten.
- **Kontoübergreifender Zugriff** – Sie können eine IAM-Rolle verwenden, um einem vertrauenswürdigen Prinzipal in einem anderen Konto den Zugriff auf Ressourcen in Ihrem Konto zu ermöglichen. Rollen stellen die primäre Möglichkeit dar, um kontoübergreifendem Zugriff zu gewähren. Bei einigen können Sie AWS-Services jedoch eine Richtlinie direkt an eine Ressource anhängen (anstatt eine Rolle als Proxy zu verwenden). Informationen zu den Unterschieden zwischen Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter [Kontoübergreifender Ressourcenzugriff in IAM](#) im IAM-Benutzerhandbuch.
- **Serviceübergreifender Zugriff** — Einige AWS-Services verwenden Funktionen in anderen AWS-Services. Wenn Sie beispielsweise in einem Service einen Anruf tätigen, ist es üblich, dass dieser Service Anwendungen in Amazon ausführt EC2 oder Objekte in Amazon S3 speichert. Ein Dienst kann dies mit den Berechtigungen des aufrufenden Prinzipals mit einer Servicerolle oder mit einer serviceverknüpften Rolle tun.
 - **Forward Access Sessions (FAS)** — Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services

könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, in Kombination mit der Anfrage, Anfragen an AWS-Service nachgelagerte Dienste zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter [Zugriffssitzungen weiterleiten](#).

- **Servicerolle** – Eine Servicerolle ist eine [IAM-Rolle](#), die ein Service übernimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter [Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service](#) im IAM-Benutzerhandbuch.
- **Dienstbezogene Rolle** — Eine dienstbezogene Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Servicebezogene Rollen erscheinen in Ihrem Dienst AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.
- **Auf Amazon ausgeführte Anwendungen EC2** — Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2 Instance ausgeführt werden und AWS API-Anfragen stellen AWS CLI . Dies ist dem Speichern von Zugriffsschlüsseln innerhalb der EC2 Instance vorzuziehen. Um einer EC2 Instanz eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein Instanzprofil, das an die Instanz angehängt ist. Ein Instanzprofil enthält die Rolle und ermöglicht Programmen, die auf der EC2 Instanz ausgeführt werden, temporäre Anmeldeinformationen abzurufen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter [Verwenden einer IAM-Rolle, um Berechtigungen für Anwendungen zu gewähren, die auf EC2 Amazon-Instances ausgeführt werden](#).

Verwalten des Zugriffs mit Richtlinien

Sie kontrollieren den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie ist ein Objekt, AWS das, wenn es einer Identität oder Ressource zugeordnet ist, deren Berechtigungen definiert. AWS wertet diese Richtlinien aus, wenn ein Prinzipal (Benutzer, Root-Benutzer oder Rollensitzung) eine Anfrage stellt. Die Berechtigungen in den Richtlinien legen fest, ob eine Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden AWS als JSON-Dokumente gespeichert. Weitere Informationen zu Struktur und

Inhalten von JSON-Richtliniendokumenten finden Sie unter [Übersicht über JSON-Richtlinien](#) im IAM-Benutzerhandbuch.

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

IAM-Richtlinien definieren Berechtigungen für eine Aktion unabhängig von der Methode, die Sie zur Ausführung der Aktion verwenden. Angenommen, es gibt eine Richtlinie, die Berechtigungen für die `iam:GetRole`-Aktion erteilt. Ein Benutzer mit dieser Richtlinie kann Rolleninformationen von der AWS Management Console AWS CLI, der oder der AWS API abrufen.

Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter [Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien](#) im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien können weiter als Inline-Richtlinien oder verwaltete Richtlinien kategorisiert werden. Inline-Richtlinien sind direkt in einen einzelnen Benutzer, eine einzelne Gruppe oder eine einzelne Rolle eingebettet. Verwaltete Richtlinien sind eigenständige Richtlinien, die Sie mehreren Benutzern, Gruppen und Rollen in Ihrem System zuordnen können AWS-Konto. Zu den verwalteten Richtlinien gehören AWS verwaltete Richtlinien und vom Kunden verwaltete Richtlinien. Informationen dazu, wie Sie zwischen einer verwalteten Richtlinie und einer Inline-Richtlinie wählen, finden Sie unter [Auswählen zwischen verwalteten und eingebundenen Richtlinien](#) im IAM-Benutzerhandbuch.

Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und

Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie [einen Prinzipal angeben](#). Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

Zugriffskontrolllisten () ACLs

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Amazon S3 und Amazon VPC sind Beispiele für Dienste, die Unterstützung ACLs bieten. AWS WAF Weitere Informationen finden Sie unter [Übersicht über ACLs die Zugriffskontrollliste \(ACL\)](#) im Amazon Simple Storage Service Developer Guide.

Weitere Richtlinientypen

AWS unterstützt zusätzliche, weniger verbreitete Richtlinientypen. Diese Richtlinientypen können die maximalen Berechtigungen festlegen, die Ihnen von den häufiger verwendeten Richtlinientypen erteilt werden können.

- **Berechtigungsgrenzen** – Eine Berechtigungsgrenze ist ein erweitertes Feature, mit der Sie die maximalen Berechtigungen festlegen können, die eine identitätsbasierte Richtlinie einer IAM-Entität (IAM-Benutzer oder -Rolle) erteilen kann. Sie können eine Berechtigungsgrenze für eine Entität festlegen. Die daraus resultierenden Berechtigungen sind der Schnittpunkt der identitätsbasierten Richtlinien einer Entität und ihrer Berechtigungsgrenzen. Ressourcenbasierte Richtlinien, die den Benutzer oder die Rolle im Feld `Principal` angeben, werden nicht durch Berechtigungsgrenzen eingeschränkt. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen über Berechtigungsgrenzen finden Sie unter [Berechtigungsgrenzen für IAM-Entitäten](#) im IAM-Benutzerhandbuch.
- **Dienststeuerungsrichtlinien (SCPs)** — SCPs sind JSON-Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OU) in festlegen. AWS Organizations AWS Organizations ist ein Dienst zur Gruppierung und zentralen Verwaltung

mehrerer Objekte AWS-Konten , die Ihrem Unternehmen gehören. Wenn Sie alle Funktionen in einer Organisation aktivieren, können Sie Richtlinien zur Servicesteuerung (SCPs) auf einige oder alle Ihre Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Root-Benutzer des AWS-Kontos Entitäten. Weitere Informationen zu Organizations und SCPs finden Sie unter [Richtlinien zur Servicesteuerung](#) im AWS Organizations Benutzerhandbuch.

- **Ressourcenkontrollrichtlinien (RCPs)** — RCPs sind JSON-Richtlinien, mit denen Sie die maximal verfügbaren Berechtigungen für Ressourcen in Ihren Konten festlegen können, ohne die IAM-Richtlinien aktualisieren zu müssen, die jeder Ressource zugeordnet sind, deren Eigentümer Sie sind. Das RCP schränkt die Berechtigungen für Ressourcen in Mitgliedskonten ein und kann sich auf die effektiven Berechtigungen für Identitäten auswirken, einschließlich der Root-Benutzer des AWS-Kontos, unabhängig davon, ob sie zu Ihrer Organisation gehören. Weitere Informationen zu Organizations RCPs, einschließlich einer Liste AWS-Services dieser Support-Leistungen RCPs, finden Sie unter [Resource Control Policies \(RCPs\)](#) im AWS Organizations Benutzerhandbuch.
- **Sitzungsrichtlinien** – Sitzungsrichtlinien sind erweiterte Richtlinien, die Sie als Parameter übergeben, wenn Sie eine temporäre Sitzung für eine Rolle oder einen verbundenen Benutzer programmgesteuert erstellen. Die resultierenden Sitzungsberechtigungen sind eine Schnittmenge der auf der Identität des Benutzers oder der Rolle basierenden Richtlinien und der Sitzungsrichtlinien. Berechtigungen können auch aus einer ressourcenbasierten Richtlinie stammen. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen finden Sie unter [Sitzungsrichtlinien](#) im IAM-Benutzerhandbuch.

Mehrere Richtlinienarten

Wenn mehrere auf eine Anforderung mehrere Richtlinienarten angewendet werden können, sind die entsprechenden Berechtigungen komplizierter. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinienarten betroffen sind, finden Sie im IAM-Benutzerhandbuch unter [Bewertungslogik für Richtlinien](#).

Wie AWS Elemental MediaTailor funktioniert mit IAM

Bevor Sie IAM zur Verwaltung des Zugriffs auf verwenden, sollten Sie sich darüber informieren MediaTailor, mit welchen IAM-Funktionen Sie arbeiten können. MediaTailor

IAM-Funktionen, die Sie mit verwenden können AWS Elemental MediaTailor

IAM-Feature	MediaTailor Unterstützung
Identitätsbasierte Richtlinien	Ja
Ressourcenbasierte Richtlinien	Ja
Richtlinienaktionen	Ja
Richtlinienressourcen	Nein
Richtlinienbedingungsschlüssel (servicespezifisch)	Ja
ACLs	Nein
ABAC (Tags in Richtlinien)	Teilweise
Temporäre Anmeldeinformationen	Ja
Prinzipalberechtigungen	Ja
Servicerollen	Nein
Serviceverknüpfte Rollen	Ja

Einen allgemeinen Überblick darüber, wie MediaTailor und andere AWS Dienste mit den meisten IAM-Funktionen funktionieren, finden Sie im [IAM-Benutzerhandbuch unter AWS Dienste, die mit IAM funktionieren](#).

Identitätsbasierte Richtlinien für MediaTailor

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter [Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien](#) im IAM-Benutzerhandbuch.

Mit identitätsbasierten IAM-Richtlinien können Sie angeben, welche Aktionen und Ressourcen zugelassen oder abgelehnt werden. Darüber hinaus können Sie die Bedingungen festlegen, unter denen Aktionen zugelassen oder abgelehnt werden. Sie können den Prinzipal nicht in einer identitätsbasierten Richtlinie angeben, da er für den Benutzer oder die Rolle gilt, dem er zugeordnet ist. Informationen zu sämtlichen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie in der [IAM-Referenz für JSON-Richtlinienelemente](#) im IAM-Benutzerhandbuch.

Beispiele für identitätsbasierte Richtlinien für MediaTailor

Beispiele für MediaTailor identitätsbasierte Richtlinien finden Sie unter. [Beispiele für identitätsbasierte Richtlinien für AWS Elemental MediaTailor](#)

Ressourcenbasierte Richtlinien finden Sie in MediaTailor

Unterstützt ressourcenbasierte Richtlinien: Ja

Der MediaTailor Dienst unterstützt nur einen Typ von ressourcenbasierten Richtlinien. Sie wird als Kanalrichtlinie bezeichnet, weil sie mit einem Kanal verknüpft ist. Diese Richtlinie definiert, welche Principals Aktionen auf dem Kanal ausführen können.

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie [einen Prinzipal angeben](#). Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Um kontoübergreifenden Zugriff zu ermöglichen, können Sie ein gesamtes Konto oder IAM-Entitäten in einem anderen Konto als Prinzipal in einer ressourcenbasierten Richtlinie angeben. Durch das Hinzufügen eines kontoübergreifenden Auftraggebers zu einer ressourcenbasierten Richtlinie ist nur die halbe Vertrauensbeziehung eingerichtet. Wenn sich der Prinzipal und die Ressource unterscheiden AWS-Konten, muss ein IAM-Administrator des vertrauenswürdigen Kontos auch der Prinzipalentität (Benutzer oder Rolle) die Berechtigung zum Zugriff auf die Ressource erteilen. Sie erteilen Berechtigungen, indem Sie der juristischen Stelle eine identitätsbasierte Richtlinie anfügen. Wenn jedoch eine ressourcenbasierte Richtlinie Zugriff auf einen Prinzipal in demselben Konto gewährt, ist keine zusätzliche identitätsbasierte Richtlinie erforderlich. Weitere Informationen finden Sie unter [Kontoübergreifender Ressourcenzugriff in IAM](#) im IAM-Benutzerhandbuch.

Informationen zum Anhängen einer ressourcenbasierten Richtlinie an einen Kanal finden Sie unter. [Erstellen Sie mit der MediaTailor Konsole einen Kanal](#)

Beispiele für ressourcenbasierte Richtlinien finden Sie MediaTailor

Beispiele für MediaTailor ressourcenbasierte Richtlinien finden Sie unter. [Beispiele für ressourcenbasierte Richtlinien für AWS Elemental MediaTailor](#)

Politische Maßnahmen für MediaTailor

Unterstützt Richtlinienaktionen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Action` einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Richtlinienaktionen haben normalerweise denselben Namen wie der zugehörige AWS API-Vorgang. Es gibt einige Ausnahmen, z. B. Aktionen, die nur mit Genehmigung durchgeführt werden können und für die es keinen passenden API-Vorgang gibt. Es gibt auch einige Operationen, die mehrere Aktionen in einer Richtlinie erfordern. Diese zusätzlichen Aktionen werden als abhängige Aktionen bezeichnet.

Schließen Sie Aktionen in eine Richtlinie ein, um Berechtigungen zur Durchführung der zugeordneten Operation zu erteilen.

Eine Liste der MediaTailor Aktionen finden Sie unter [Aktionen definiert von AWS Elemental MediaTailor](#) in der Serviceautorisierungsreferenz.

Bei Richtlinienaktionen wird vor der Aktion das folgende Präfix MediaTailor verwendet:

```
mediatailor
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Kommata:

```
"Action": [  
    "mediatailor:action1",  
    "mediatailor:action2"
```

]

Beispiele für MediaTailor identitätsbasierte Richtlinien finden Sie unter. [Beispiele für identitätsbasierte Richtlinien für AWS Elemental MediaTailor](#)

Politische Ressourcen für MediaTailor

Unterstützt politische Ressourcen: Nein

AWS Elemental MediaTailor unterstützt die Angabe von Ressourcen ARNs in einer Richtlinie nicht.

Schlüssel zur Richtlinienbedingung für MediaTailor

Unterstützt servicespezifische Richtlinienbedingungsschlüssel: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das Element `Condition` (oder `Condition block`) ermöglicht Ihnen die Angabe der Bedingungen, unter denen eine Anweisung wirksam ist. Das Element `Condition` ist optional. Sie können bedingte Ausdrücke erstellen, die [Bedingungsoperatoren](#) verwenden, z. B. ist gleich oder kleiner als, damit die Bedingung in der Richtlinie mit Werten in der Anforderung übereinstimmt.

Wenn Sie mehrere `Condition`-Elemente in einer Anweisung oder mehrere Schlüssel in einem einzelnen `Condition`-Element angeben, wertet AWS diese mittels einer logischen AND-Operation aus. Wenn Sie mehrere Werte für einen einzelnen Bedingungsschlüssel angeben, AWS wertet die Bedingung mithilfe einer logischen OR Operation aus. Alle Bedingungen müssen erfüllt werden, bevor die Berechtigungen der Anweisung gewährt werden.

Sie können auch Platzhaltervariablen verwenden, wenn Sie Bedingungen angeben. Beispielsweise können Sie einem IAM-Benutzer die Berechtigung für den Zugriff auf eine Ressource nur dann gewähren, wenn sie mit dessen IAM-Benutzernamen gekennzeichnet ist. Weitere Informationen finden Sie unter [IAM-Richtlinienelemente: Variablen und Tags](#) im IAM-Benutzerhandbuch.

AWS unterstützt globale Bedingungsschlüssel und dienstspezifische Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter [Kontextschlüssel für AWS globale Bedingungen](#) im IAM-Benutzerhandbuch.

Eine Liste der MediaTailor Bedingungsschlüssel finden Sie unter [Bedingungsschlüssel für AWS Elemental MediaTailor](#) in der Service Authorization Reference. Informationen zu den Aktionen und Ressourcen, mit denen Sie einen Bedingungsschlüssel verwenden können, finden Sie unter [Aktionen definiert von AWS Elemental MediaTailor](#).

AWS Elemental MediaTailor stellt keine dienstspezifischen Bedingungsschlüssel bereit, unterstützt aber die Verwendung einiger globaler Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter [AWS Globale Bedingungskontextschlüssel](#) im AWS Identity and Access Management Benutzerhandbuch.

ACLs in MediaTailor

Unterstützt ACLs: Nein

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

ABAC mit MediaTailor

Unterstützt ABAC (Tags in Richtlinien): Teilweise

Die attributbasierte Zugriffskontrolle (ABAC) ist eine Autorisierungsstrategie, bei der Berechtigungen basierend auf Attributen definiert werden. In AWS werden diese Attribute als Tags bezeichnet. Sie können Tags an IAM-Entitäten (Benutzer oder Rollen) und an viele AWS Ressourcen anhängen. Das Markieren von Entitäten und Ressourcen ist der erste Schritt von ABAC. Anschließend entwerfen Sie ABAC-Richtlinien, um Operationen zuzulassen, wenn das Tag des Prinzipals mit dem Tag der Ressource übereinstimmt, auf die sie zugreifen möchten.

ABAC ist in Umgebungen hilfreich, die schnell wachsen, und unterstützt Sie in Situationen, in denen die Richtlinienverwaltung mühsam wird.

Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungelement einer [Richtlinie Tag-Informationen](#) an, indem Sie die Schlüssel `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, oder Bedingung `aws:TagKeys` verwenden.

Wenn ein Service alle drei Bedingungsschlüssel für jeden Ressourcentyp unterstützt, lautet der Wert für den Service Ja. Wenn ein Service alle drei Bedingungsschlüssel für nur einige Ressourcentypen unterstützt, lautet der Wert Teilweise.

Weitere Informationen zu ABAC finden Sie unter [Definieren von Berechtigungen mit ABAC-Autorisierung](#) im IAM-Benutzerhandbuch. Um ein Tutorial mit Schritten zur Einstellung von ABAC anzuzeigen, siehe [Attributbasierte Zugriffskontrolle \(ABAC\)](#) verwenden im IAM-Benutzerhandbuch.

Verwenden Sie für MediaTailor den Wert Partial.

Verwenden temporärer Anmeldeinformationen mit MediaTailor

Unterstützt temporäre Anmeldeinformationen: Ja

Einige funktionieren AWS-Services nicht, wenn Sie sich mit temporären Anmeldeinformationen anmelden. Weitere Informationen, einschließlich Informationen, die mit temporären Anmeldeinformationen AWS-Services [funktionieren AWS-Services](#) , finden Sie im [IAM-Benutzerhandbuch unter Diese Option funktioniert mit IAM](#).

Sie verwenden temporäre Anmeldeinformationen, wenn Sie sich mit einer anderen AWS Management Console Methode als einem Benutzernamen und einem Passwort anmelden. Wenn Sie beispielsweise AWS über den Single Sign-On-Link (SSO) Ihres Unternehmens darauf zugreifen, werden bei diesem Vorgang automatisch temporäre Anmeldeinformationen erstellt. Sie erstellen auch automatisch temporäre Anmeldeinformationen, wenn Sie sich als Benutzer bei der Konsole anmelden und dann die Rollen wechseln. Weitere Informationen zum Wechseln von Rollen finden Sie unter [Wechseln von einer Benutzerrolle zu einer IAM-Rolle \(Konsole\)](#) im IAM-Benutzerhandbuch.

Mithilfe der AWS API AWS CLI oder können Sie temporäre Anmeldeinformationen manuell erstellen. Sie können diese temporären Anmeldeinformationen dann für den Zugriff verwenden AWS. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter [Temporäre Sicherheitsanmeldeinformationen in IAM](#).

Serviceübergreifende Prinzipalberechtigungen für MediaTailor

Unterstützt Forward Access Sessions (FAS): Ja

Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, kombiniert mit der Anforderung, Anfragen an nachgelagerte Dienste AWS-Service zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem

Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter [Zugriffssitzungen weiterleiten](#).

Servicerollen für MediaTailor

Unterstützt Servicerollen: Nein

AWS Elemental MediaTailor unterstützt keine Servicerollen.

Mit Diensten verknüpfte Rollen für MediaTailor

Unterstützt dienstbezogene Rollen: Ja

Eine serviceverknüpfte Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Dienstbezogene Rollen werden in Ihrem Dienst angezeigt AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

Einzelheiten zum Erstellen oder Verwalten von MediaTailor dienstbezogenen Rollen finden Sie unter [Verwenden von serviceverknüpften Rollen für MediaTailor](#)

Beispiele für identitätsbasierte Richtlinien für AWS Elemental MediaTailor

Benutzer und Rollen haben standardmäßig nicht die Berechtigung, MediaTailor-Ressourcen zu erstellen oder zu ändern. Sie können auch keine Aufgaben mithilfe der AWS API AWS Management Console, AWS Command Line Interface (AWS CLI) oder ausführen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter [Erstellen von IAM-Richtlinien \(Konsole\)](#) im IAM-Benutzerhandbuch.

Einzelheiten zu Aktionen und Ressourcentypen, die von definiert wurden MediaTailor, einschließlich des Formats von ARNs für jeden der Ressourcentypen, finden Sie unter [Aktionen, Ressourcen und Bedingungsschlüssel für AWS Elemental MediaTailor](#) in der Service Authorization Reference.

Themen

- [Bewährte Methoden für Richtlinien](#)

- [Verwenden der MediaTailor-Konsole](#)
- [Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer](#)

Bewährte Methoden für Richtlinien

Identitätsbasierte Richtlinien legen fest, ob jemand MediaTailor Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder sie löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen und Empfehlungen:

- Beginnen Sie mit AWS verwalteten Richtlinien und wechseln Sie zu Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um Ihren Benutzern und Workloads zunächst Berechtigungen zu gewähren. Sie sind in Ihrem verfügbar. AWS-Konto Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom AWS Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind. Weitere Informationen finden Sie unter [AWS -verwaltete Richtlinien](#) oder [AWS -verwaltete Richtlinien für Auftrags-Funktionen](#) im IAM-Benutzerhandbuch.
- Anwendung von Berechtigungen mit den geringsten Rechten – Wenn Sie mit IAM-Richtlinien Berechtigungen festlegen, gewähren Sie nur die Berechtigungen, die für die Durchführung einer Aufgabe erforderlich sind. Sie tun dies, indem Sie die Aktionen definieren, die für bestimmte Ressourcen unter bestimmten Bedingungen durchgeführt werden können, auch bekannt als die geringsten Berechtigungen. Weitere Informationen zur Verwendung von IAM zum Anwenden von Berechtigungen finden Sie unter [Richtlinien und Berechtigungen in IAM](#) im IAM-Benutzerhandbuch.
- Verwenden von Bedingungen in IAM-Richtlinien zur weiteren Einschränkung des Zugriffs – Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen zu beschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben, um festzulegen, dass alle Anforderungen mithilfe von SSL gesendet werden müssen. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese für einen bestimmten Zweck verwendet werden AWS-Service, z. AWS CloudFormation B. Weitere Informationen finden Sie unter [IAM-JSON-Richtlinienelemente: Bedingung](#) im IAM-Benutzerhandbuch.
- Verwenden von IAM Access Analyzer zur Validierung Ihrer IAM-Richtlinien, um sichere und funktionale Berechtigungen zu gewährleisten – IAM Access Analyzer validiert neue und vorhandene Richtlinien, damit die Richtlinien der IAM-Richtliniensprache (JSON)

und den bewährten IAM-Methoden entsprechen. IAM Access Analyzer stellt mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen zur Verfügung, damit Sie sichere und funktionale Richtlinien erstellen können. Weitere Informationen finden Sie unter [Richtlinienvvalidierung mit IAM Access Analyzer](#) im IAM-Benutzerhandbuch.

- Multi-Faktor-Authentifizierung (MFA) erforderlich — Wenn Sie ein Szenario haben, das IAM-Benutzer oder einen Root-Benutzer in Ihrem System erfordert AWS-Konto, aktivieren Sie MFA für zusätzliche Sicherheit. Um MFA beim Aufrufen von API-Vorgängen anzufordern, fügen Sie Ihren Richtlinien MFA-Bedingungen hinzu. Weitere Informationen finden Sie unter [Sicherer API-Zugriff mit MFA](#) im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter [Bewährte Methoden für die Sicherheit in IAM](#) im IAM-Benutzerhandbuch.

Verwenden der MediaTailor-Konsole

Um auf die AWS Elemental MediaTailor Konsole zugreifen zu können, benötigen Sie ein Mindestmaß an Berechtigungen. Diese Berechtigungen müssen es Ihnen ermöglichen, Details zu den MediaTailor Ressourcen in Ihrem aufzulisten und anzuzeigen AWS-Konto. Wenn Sie eine identitätsbasierte Richtlinie erstellen, die strenger ist als die mindestens erforderlichen Berechtigungen, funktioniert die Konsole nicht wie vorgesehen für Entitäten (Benutzer oder Rollen) mit dieser Richtlinie.

Sie müssen Benutzern, die nur die API AWS CLI oder die AWS API aufrufen, keine Mindestberechtigungen für die Konsole gewähren. Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die der API-Operation entsprechen, die die Benutzer ausführen möchten.

Um sicherzustellen, dass Benutzer und Rollen die MediaTailor Konsole weiterhin verwenden können, fügen Sie den Entitäten auch die MediaTailor *ConsoleAccess* oder die *ReadOnly* AWS verwaltete Richtlinie hinzu. Weitere Informationen finden Sie unter [Hinzufügen von Berechtigungen zu einem Benutzer](#) im IAM-Benutzerhandbuch.

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die IAM-Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer Benutzeridentität angefügt sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der AWS CLI AWS OR-API.

```
{
```

```

"Version": "2012-10-17",
"Statement": [
  {
    "Sid": "ViewOwnUserInfo",
    "Effect": "Allow",
    "Action": [
      "iam:GetUserPolicy",
      "iam:ListGroupsWithUser",
      "iam:ListAttachedUserPolicies",
      "iam:ListUserPolicies",
      "iam:GetUser"
    ],
    "Resource": ["arn:aws:iam::*:user/${aws:username}"]
  },
  {
    "Sid": "NavigateInConsole",
    "Effect": "Allow",
    "Action": [
      "iam:GetGroupPolicy",
      "iam:GetPolicyVersion",
      "iam:GetPolicy",
      "iam:ListAttachedGroupPolicies",
      "iam:ListGroupPolicies",
      "iam:ListPolicyVersions",
      "iam:ListPolicies",
      "iam:ListUsers"
    ],
    "Resource": "*"
  }
]
}

```

Beispiele für ressourcenbasierte Richtlinien für AWS Elemental MediaTailor

Informationen zum Anhängen einer ressourcenbasierten Richtlinie an einen Channel finden Sie unter [Erstellen Sie mit der MediaTailor Konsole einen Kanal](#)

Themen

- [Anonymer Zugriff](#)
- [Kontoübergreifender Zugriff](#)

Anonymer Zugriff

Beachten Sie die folgende Allow Richtlinie. Wenn diese Richtlinie in Kraft ist, wird anonymer Zugriff auf die `mediatailor:GetManifest` Aktion auf der Kanalressource in der Richtlinie MediaTailor ermöglicht. Dies geschieht, wenn *region* der AWS-Region, *accountID* Ihre AWS-Konto ID und der Name der Kanalressource *channelName* ist.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAnonymous",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "mediatailor:GetManifest",
      "Resource": "arn:aws:mediatailor:region:accountID:channel/channelName"
    }
  ]
}
```

Kontoübergreifender Zugriff

Beachten Sie die folgende Allow Richtlinie. Wenn diese Richtlinie in Kraft ist, MediaTailor ist die `mediatailor:GetManifest` Aktion für die Kanalressource in der Richtlinie kontenübergreifend möglich. Dies geschieht, wenn *region* der AWS-Region, *accountID* deine AWS-Konto ID und der Name der Kanalressource *channelName* ist.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCrossAccountAccess",
      "Effect": "Allow",
      "Principal": {"AWS": "arn:aws:iam::111111111111:root"},
      "Action": "mediatailor:GetManifest",
      "Resource": "arn:aws:mediatailor:region:accountID:channel/channelName"
    }
  ]
}
```

AWS verwaltete Richtlinien für AWS Elemental MediaTailor

Um Benutzern, Gruppen und Rollen Berechtigungen hinzuzufügen, ist es einfacher, AWS verwaltete Richtlinien zu verwenden, als Richtlinien selbst zu schreiben. Es erfordert Zeit und Fachwissen, um [von Kunden verwaltete IAM-Richtlinien zu erstellen](#), die Ihrem Team nur die benötigten Berechtigungen bieten. Um schnell loszulegen, können Sie unsere AWS verwalteten Richtlinien verwenden. Diese Richtlinien decken allgemeine Anwendungsfälle ab und sind in Ihrem AWS-Konto verfügbar. Weitere Informationen zu AWS verwalteten Richtlinien finden Sie im IAM-Benutzerhandbuch unter [AWS Verwaltete Richtlinien](#).

AWS Dienste verwalten und aktualisieren AWS verwaltete Richtlinien. Sie können die Berechtigungen in AWS verwalteten Richtlinien nicht ändern. Services fügen einer von AWS verwalteten Richtlinien gelegentlich zusätzliche Berechtigungen hinzu, um neue Features zu unterstützen. Diese Art von Update betrifft alle Identitäten (Benutzer, Gruppen und Rollen), an welche die Richtlinie angehängt ist. Services aktualisieren eine von AWS verwaltete Richtlinie am ehesten, ein neues Feature gestartet wird oder neue Vorgänge verfügbar werden. Dienste entfernen keine Berechtigungen aus einer AWS verwalteten Richtlinie, sodass durch Richtlinienaktualisierungen Ihre bestehenden Berechtigungen nicht beeinträchtigt werden.

AWS Unterstützt außerdem verwaltete Richtlinien für Jobfunktionen, die sich über mehrere Dienste erstrecken. Die ReadOnlyAccess AWS verwaltete Richtlinie bietet beispielsweise schreibgeschützten Zugriff auf alle AWS Dienste und Ressourcen. Wenn ein Dienst eine neue Funktion startet, werden nur Leseberechtigungen für neue Operationen und Ressourcen AWS hinzugefügt. Eine Liste und Beschreibungen der Richtlinien für Auftragsfunktionen finden Sie in [Verwaltete AWS -Richtlinien für Auftragsfunktionen](#) im IAM-Leitfaden.

AWS verwaltete Richtlinie: AWSElementalMediaTailorFullAccess

Sie können die AWSElementalMediaTailorFullAccess-Richtlinie an Ihre IAM-Identitäten anfügen. Dies ist nützlich für Benutzer, die Wiedergabekonfigurationen und Ressourcen zur Kanalzusammenstellung wie Programme und Kanäle erstellen und verwalten müssen. Diese Richtlinie gewährt Berechtigungen, die vollen Zugriff auf ermöglichen AWS Elemental MediaTailor. Diese Benutzer können MediaTailor Ressourcen erstellen, aktualisieren und löschen.

Informationen zu den Berechtigungen für diese Richtlinie finden Sie unter [AWSElementalMediaTailorFullAccess](#) in der Referenz zu von AWS verwalteten Richtlinien.

AWS verwaltete Richtlinie: AWSElementalMediaTailorReadOnly

Sie können die `AWSElementalMediaTailorReadOnly`-Richtlinie an Ihre IAM-Identitäten anfügen. Dies ist nützlich für Benutzer, die Wiedergabekonfigurationen und Ressourcen zur Kanalzusammenstellung wie Programme und Kanäle einsehen müssen. Diese Richtlinie gewährt Berechtigungen, die nur Lesezugriff auf ermöglichen. AWS Elemental MediaTailor Diese Benutzer können MediaTailor -Ressourcen nicht erstellen, aktualisieren oder löschen.

Informationen zu den Berechtigungen für diese Richtlinie finden Sie unter [AWSElementalMediaTailorReadOnly](#) in der Referenz zu von AWS verwalteten Richtlinien.

MediaTailor Aktualisierungen verwalteter Richtlinien AWS

Hier finden Sie Informationen zu Aktualisierungen der AWS verwalteten Richtlinien, die MediaTailor seit Beginn der Nachverfolgung dieser Änderungen durch diesen Dienst vorgenommen wurden. Um automatische Warnungen über Änderungen an dieser Seite erhalten, abonnieren Sie den RSS-Feed auf MediaTailor [Dokumenthistorie für AWS Elemental MediaTailor](#).

Änderung	Beschreibung	Datum
MediaTailor neue verwaltete Richtlinien hinzugefügt	MediaTailor hat die folgenden verwalteten Richtlinien hinzugefügt: • AWSElementalMediaTailorReadOnly • AWSElementalMediaTailorFullAccess	24. November 2021
MediaTailor hat begonnen, Änderungen zu verfolgen	MediaTailor hat begonnen, Änderungen für die AWS verwalteten Richtlinien zu verfolgen.	24. November 2021

Verwenden von serviceverknüpften Rollen für MediaTailor

AWS Elemental MediaTailor verwendet AWS Identity and Access Management (IAM) [serviceverknüpfte](#) Rollen. Eine serviceverknüpfte Rolle ist ein einzigartiger Typ von IAM-Rolle, mit

der direkt verknüpft ist. MediaTailor Mit Diensten verknüpfte Rollen sind vordefiniert MediaTailor und enthalten alle Berechtigungen, die der Dienst benötigt, um andere AWS Dienste in Ihrem Namen aufzurufen.

Eine dienstbezogene Rolle MediaTailor erleichtert die Einrichtung, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. MediaTailor definiert die Berechtigungen ihrer dienstbezogenen Rollen und MediaTailor kann, sofern nicht anders definiert, nur ihre Rollen übernehmen. Die definierten Berechtigungen umfassen die Vertrauens- und Berechtigungsrichtlinie. Diese Berechtigungsrichtlinie kann keinen anderen IAM-Entitäten zugewiesen werden.

Sie können eine serviceverknüpfte Rolle erst löschen, nachdem ihre verwandten Ressourcen gelöscht wurden. Dadurch werden Ihre MediaTailor Ressourcen geschützt, da Sie nicht versehentlich die Zugriffsberechtigung für die Ressourcen entziehen können.

Informationen zu anderen Services, die serviceverknüpften Rollen unterstützen, finden Sie unter [AWS -Services, die mit IAM funktionieren](#). Suchen Sie nach den Services, für die Ja in der Spalte Serviceverknüpfte Rolle angegeben ist. Wählen Sie über einen Link Ja aus, um die Dokumentation zu einer serviceverknüpften Rolle für diesen Service anzuzeigen.

Unterstützte Regionen für serviceverknüpfte Rollen MediaTailor

MediaTailor unterstützt die Verwendung von dienstbezogenen Rollen in allen Regionen, in denen der Dienst verfügbar ist. Weitere Informationen finden Sie unter [AWS -Regionen und Endpunkte](#).

Themen

- [Dienstbezogene Rollenberechtigungen für MediaTailor](#)
- [Erstellen einer dienstbezogenen Rolle für MediaTailor](#)
- [Bearbeiten einer dienstbezogenen Rolle für MediaTailor](#)
- [Löschen einer dienstbezogenen Rolle für MediaTailor](#)

Dienstbezogene Rollenberechtigungen für MediaTailor

MediaTailor verwendet die mit dem Dienst verknüpfte Rolle mit dem Namen `AWSServiceRoleForMediaTailor`— MediaTailor verwendet diese dienstverknüpfte Rolle, um Protokollgruppen, Protokolldatenströme und Protokollereignisse aufzurufen und CloudWatch zu verwalten. Diese verwaltete Richtlinie ist mit der folgenden serviceverknüpften Rolle verbunden: `AWSMediaTailorServiceRolePolicy`.

Die AWSService RoleForMediaTailor dienstverknüpfte Rolle vertraut darauf, dass die folgenden Dienste die Rolle übernehmen:

- `mediatailor.amazonaws.com`

Die Richtlinie für Rollenberechtigungen MediaTailor ermöglicht es, die folgenden Aktionen für die angegebenen Ressourcen durchzuführen:

- Aktion: `logs:PutLogEvents` für `arn:aws:logs:*:*:log-group:/aws/MediaTailor/*:log-stream:*`
- Aktion: `logs:CreateLogStream`, `logs:CreateLogGroup`, `logs:DescribeLogGroups`, `logs:DescribeLogStreams` für `arn:aws:logs:*:*:log-group:/aws/MediaTailor/*`

Sie müssen Berechtigungen konfigurieren, damit eine juristische Stelle von IAM (z. B. Benutzer, Gruppe oder Rolle) eine serviceverknüpfte Rolle erstellen, bearbeiten oder löschen kann. Weitere Informationen finden Sie unter [serviceverknüpfte Rollenberechtigungen](#) im IAM-Benutzerhandbuch.

Erstellen einer dienstbezogenen Rolle für MediaTailor

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. Wenn Sie die Sitzungsprotokollierung in der AWS Management Console, der AWS Command Line Interface (AWS CLI) oder der AWS API aktivieren, MediaTailor wird die dienstbezogene Rolle für Sie erstellt.

Important

Diese serviceverknüpfte Rolle kann in Ihrem Konto erscheinen, wenn Sie eine Aktion in einem anderen Service abgeschlossen haben, der die von dieser Rolle unterstützten Features verwendet. Außerdem gilt: Wenn Sie den MediaTailor Dienst vor dem 15. September 2021 genutzt haben, als er begann, dienstbezogene Rollen zu unterstützen, haben Sie die AWSService RoleForMediaTailor Rolle dann in Ihrem Konto MediaTailor erstellt. Weitere Informationen finden Sie unter [Eine neue Rolle ist in meinem IAM-Konto erschienen](#).

Wenn Sie diese serviceverknüpfte Rolle löschen und sie dann erneut erstellen müssen, können Sie dasselbe Verfahren anwenden, um die Rolle in Ihrem Konto neu anzulegen. Wenn Sie die Sitzungsprotokollierung aktivieren, MediaTailor wird die dienstbezogene Rolle erneut für Sie erstellt.

Sie können auch die IAM-Konsole verwenden, um eine serviceverknüpfte Rolle mit dem MediaTailorAnwendungsfall zu erstellen. Erstellen Sie in der AWS CLI oder der AWS API eine dienstverknüpfte Rolle mit dem `mediatailor.amazonaws.com` Dienstenamen. Weitere Informationen finden Sie unter [Erstellen einer serviceverknüpfte Rolle](#) im IAM-Leitfaden. Wenn Sie diese serviceverknüpfte Rolle löschen, können Sie mit demselben Verfahren die Rolle erneut erstellen.

Bearbeiten einer dienstbezogenen Rolle für MediaTailor

MediaTailor erlaubt es Ihnen nicht, die `AWSServiceRoleForMediaTailor` dienstbezogene Rolle zu bearbeiten. Da möglicherweise verschiedene Entitäten auf die Rolle verweisen, kann der Rollename nach dem Erstellen einer serviceverknüpften Rolle nicht mehr geändert werden. Sie können jedoch die Beschreibung der Rolle mit IAM bearbeiten. Weitere Informationen finden Sie unter [Bearbeiten einer serviceverknüpften Rolle](#) im IAM-Benutzerhandbuch.

Löschen einer dienstbezogenen Rolle für MediaTailor

Wenn Sie ein Feature oder einen Dienst, die bzw. der eine serviceverknüpften Rolle erfordert, nicht mehr benötigen, sollten Sie diese Rolle löschen. Auf diese Weise haben Sie keine ungenutzte juristische Stelle, die nicht aktiv überwacht oder verwaltet wird. Sie müssen jedoch die Ressourcen für Ihre serviceverknüpften Rolle zunächst bereinigen, bevor Sie sie manuell löschen können.

Note

Wenn der MediaTailor Dienst die Rolle verwendet, wenn Sie versuchen, die Ressourcen zu bereinigen, schlägt das Löschen möglicherweise fehl. Wenn dies passiert, warten Sie einige Minuten und versuchen Sie es erneut.

Um MediaTailor Ressourcen zu bereinigen, die von der `AWSServiceRoleForMediaTailor`

- Bevor Sie die dienstbezogene Rolle löschen können, die von MediaTailor für die Protokollkonfiguration erstellt wurde, müssen Sie zunächst alle Protokollkonfigurationen in Ihrem Konto deaktivieren. Um eine Protokollkonfiguration zu deaktivieren, setzen Sie den Wert „Prozent aktiviert“ auf 0. Dadurch wird die gesamte Sitzungsprotokollierung der entsprechenden Wiedergabekonfiguration deaktiviert. Weitere Informationen finden Sie unter [Deaktivierung einer Protokollkonfiguration](#).

So löschen Sie die serviceverknüpfte Rolle mit IAM

Verwenden Sie die IAM-Konsole, die AWS Command Line Interface (AWS CLI) oder die AWS API, um die `AWSServiceRoleForMediaTailor` serviceverknüpfte Rolle zu löschen. Weitere Informationen finden Sie unter [Löschen einer serviceverknüpften Rolle](#) im IAM-Leitfaden.

Fehlerbehebung bei AWS Elemental MediaTailor Identität und Zugriff

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit MediaTailor und IAM auftreten können.

Themen

- [Ich bin nicht berechtigt, eine Aktion durchzuführen in MediaTailor](#)
- [Ich bin nicht berechtigt, iam auszuführen: PassRole](#)
- [Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine MediaTailor Ressourcen ermöglichen](#)

Ich bin nicht berechtigt, eine Aktion durchzuführen in MediaTailor

Wenn Sie eine Fehlermeldung erhalten, dass Sie nicht zur Durchführung einer Aktion berechtigt sind, müssen Ihre Richtlinien aktualisiert werden, damit Sie die Aktion durchführen können.

Der folgende Beispielfehler tritt auf, wenn der IAM-Benutzer `mateojackson` versucht, über die Konsole Details zu einer fiktiven *my-example-widget*-Ressource anzuzeigen, jedoch nicht über `mediatailor:GetWidget`-Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
mediatailor:GetWidget on resource: my-example-widget
```

In diesem Fall muss die Richtlinie für den Benutzer `mateojackson` aktualisiert werden, damit er mit der `mediatailor:GetWidget`-Aktion auf die *my-example-widget*-Ressource zugreifen kann.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich bin nicht berechtigt, iam auszuführen: PassRole

Wenn Sie die Fehlermeldung erhalten, dass Sie nicht zum Durchführen der `iam:PassRole`-Aktion autorisiert sind, müssen Ihre Richtlinien aktualisiert werden, um eine Rolle an MediaTailor übergeben zu können.

Einige AWS-Services ermöglichen es Ihnen, eine bestehende Rolle an diesen Dienst zu übergeben, anstatt eine neue Servicerolle oder eine dienstverknüpfte Rolle zu erstellen. Hierzu benötigen Sie Berechtigungen für die Übergabe der Rolle an den Dienst.

Der folgende Beispielfehler tritt auf, wenn ein IAM-Benutzer mit dem Namen `marymajor` versucht, die Konsole zu verwenden, um eine Aktion in MediaTailor auszuführen. Die Aktion erfordert jedoch, dass der Service über Berechtigungen verfügt, die durch eine Servicerolle gewährt werden. Mary besitzt keine Berechtigungen für die Übergabe der Rolle an den Dienst.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

In diesem Fall müssen die Richtlinien von Mary aktualisiert werden, um die Aktion `iam:PassRole` ausführen zu können.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine MediaTailor Ressourcen ermöglichen

Sie können eine Rolle erstellen, die Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation für den Zugriff auf Ihre Ressourcen verwenden können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Für Dienste, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (ACLs) unterstützen, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob diese Funktionen MediaTailor unterstützt werden, finden Sie unter [Wie AWS Elemental MediaTailor funktioniert mit IAM](#)
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie im IAM-Benutzerhandbuch unter [Gewähren des Zugriffs auf einen IAM-Benutzer in einem anderen AWS-Konto , den Sie besitzen](#).
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie [AWS-Konten im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für Dritte](#).
- Informationen dazu, wie Sie über einen Identitätsverbund Zugriff gewähren, finden Sie unter [Gewähren von Zugriff für extern authentifizierte Benutzer \(Identitätsverbund\)](#) im IAM-Benutzerhandbuch.

- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter [Kontoübergreifender Ressourcenzugriff in IAM](#) im IAM-Benutzerhandbuch.

Konformitätsprüfung für AWS Elemental MediaTailor

Informationen darüber, ob AWS-Service ein [AWS-Services in den Geltungsbereich bestimmter Compliance-Programme fällt](#), finden Sie unter [Umfang nach Compliance-Programm AWS-Services unter](#) . Wählen Sie dort das Compliance-Programm aus, an dem Sie interessiert sind. Allgemeine Informationen finden Sie unter [AWS Compliance-Programme AWS](#) .

Sie können Prüfberichte von Drittanbietern unter herunterladen AWS Artifact. Weitere Informationen finden Sie unter [Berichte herunterladen unter](#) .

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services hängt von der Vertraulichkeit Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. AWS stellt die folgenden Ressourcen zur Verfügung, die Sie bei der Einhaltung der Vorschriften unterstützen:

- [Compliance und Governance im Bereich Sicherheit](#) – In diesen Anleitungen für die Lösungsimplementierung werden Überlegungen zur Architektur behandelt. Außerdem werden Schritte für die Bereitstellung von Sicherheits- und Compliance-Features beschrieben.
- [Referenz für berechnete HIPAA-Services](#) – Listet berechnete HIPAA-Services auf. Nicht alle AWS-Services sind HIPAA-fähig.
- [AWS Compliance-Ressourcen](#) — Diese Sammlung von Arbeitsmappen und Leitfäden gilt möglicherweise für Ihre Branche und Ihren Standort.
- [AWS Leitfäden zur Einhaltung von Vorschriften für Kunden](#) — Verstehen Sie das Modell der gemeinsamen Verantwortung aus dem Blickwinkel der Einhaltung von Vorschriften. In den Leitfäden werden die bewährten Verfahren zur Sicherung zusammengefasst AWS-Services und die Leitlinien den Sicherheitskontrollen in verschiedenen Frameworks (einschließlich des National Institute of Standards and Technology (NIST), des Payment Card Industry Security Standards Council (PCI) und der International Organization for Standardization (ISO)) zugeordnet.
- [Evaluierung von Ressourcen anhand von Regeln](#) im AWS Config Entwicklerhandbuch — Der AWS Config Service bewertet, wie gut Ihre Ressourcenkonfigurationen den internen Praktiken, Branchenrichtlinien und Vorschriften entsprechen.

- [AWS Security Hub](#)— Auf diese AWS-Service Weise erhalten Sie einen umfassenden Überblick über Ihren internen Sicherheitsstatus. AWS Security Hub verwendet Sicherheitskontrollen, um Ihre AWS -Ressourcen zu bewerten und Ihre Einhaltung von Sicherheitsstandards und bewährten Methoden zu überprüfen. Die Liste der unterstützten Services und Kontrollen finden Sie in der [Security-Hub-Steuerelementreferenz](#).
- [Amazon GuardDuty](#) — Dies AWS-Service erkennt potenzielle Bedrohungen für Ihre Workloads AWS-Konten, Container und Daten, indem es Ihre Umgebung auf verdächtige und böswillige Aktivitäten überwacht. GuardDuty kann Ihnen helfen, verschiedene Compliance-Anforderungen wie PCI DSS zu erfüllen, indem es die in bestimmten Compliance-Frameworks vorgeschriebenen Anforderungen zur Erkennung von Eindringlingen erfüllt.
- [AWS Audit Manager](#)— Auf diese AWS-Service Weise können Sie Ihre AWS Nutzung kontinuierlich überprüfen, um das Risikomanagement und die Einhaltung von Vorschriften und Industriestandards zu vereinfachen.

Resilienz in AWS Elemental MediaTailor

Die AWS globale Infrastruktur basiert auf AWS-Regionen Availability Zones. AWS-Regionen bieten mehrere physisch getrennte und isolierte Availability Zones, die über Netzwerke mit niedriger Latenz, hohem Durchsatz und hoher Redundanz miteinander verbunden sind. Mithilfe von Availability Zones können Sie Anwendungen und Datenbanken erstellen und ausführen, die automatisch Failover zwischen Zonen ausführen, ohne dass es zu Unterbrechungen kommt. Availability Zones sind besser verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Weitere Informationen zu Availability Zones AWS-Regionen und Availability Zones finden Sie unter [AWS Globale](#) Infrastruktur.

Zusätzlich zur AWS globalen Infrastruktur MediaTailor bietet es mehrere Funktionen zur Unterstützung Ihrer Datenausfallsicherheit und Backup-Anforderungen.

Sicherheit der Infrastruktur in MediaTailor

Als verwalteter Dienst AWS Elemental MediaTailor ist er durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS Sicherheitsdiensten und zum AWS Schutz der Infrastruktur finden Sie unter [AWS Cloud-Sicherheit](#). Informationen zum Entwerfen Ihrer AWS Umgebung unter Verwendung der bewährten Methoden für die Infrastruktursicherheit finden Sie unter [Infrastructure Protection](#) in Security Pillar AWS Well-Architected Framework.

Sie verwenden AWS veröffentlichte API-Aufrufe für den Zugriff MediaTailor über das Netzwerk. Kunden müssen Folgendes unterstützen:

- Transport Layer Security (TLS). Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Verschlüsselungs-Suiten mit Perfect Forward Secrecy (PFS) wie DHE (Ephemeral Diffie-Hellman) oder ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi.

Außerdem müssen Anforderungen mit einer Zugriffsschlüssel-ID und einem geheimen Zugriffsschlüssel signiert sein, der einem IAM-Prinzipal zugeordnet ist. Alternativ können Sie mit [AWS Security Token Service](#) (AWS STS) temporäre Sicherheitsanmeldeinformationen erstellen, um die Anforderungen zu signieren.

Sie können diese API-Operationen von jedem Netzwerkstandort aus aufrufen, MediaTailor unterstützt jedoch ressourcenbasierte Zugriffsrichtlinien, die Einschränkungen auf der Grundlage der Quell-IP-Adresse beinhalten können. Sie können auch MediaTailor Richtlinien verwenden, um den Zugriff von bestimmten Amazon Virtual Private Cloud (Amazon VPC) -Endpunkten oder bestimmten zu kontrollieren. VPCs Dadurch wird der Netzwerkzugriff auf eine bestimmte MediaTailor Ressource effektiv nur von der spezifischen VPC innerhalb des AWS Netzwerks isoliert.

Serviceübergreifende Confused-Deputy-Prävention

Das Problem des verwirrten Stellvertreters ist ein Sicherheitsproblem, bei dem eine Entität, die keine Berechtigung zur Durchführung einer Aktion hat, eine privilegiertere Entität zur Durchführung der Aktion zwingen kann. In AWS kann ein dienstübergreifendes Identitätswechsels zu einem Problem mit dem verwirrten Stellvertreter führen. Ein dienstübergreifender Identitätswechsel kann auftreten, wenn ein Dienst (der Anruf-Dienst) einen anderen Dienst anruft (den aufgerufenen Dienst). Der aufrufende Service kann manipuliert werden, um seine Berechtigungen zu verwenden, um Aktionen auf die Ressourcen eines anderen Kunden auszuführen, für die er sonst keine Zugriffsberechtigung haben sollte. Um dies zu verhindern, bietet AWS Tools, mit denen Sie Ihre Daten für alle Services mit Serviceprinzipalen schützen können, die Zugriff auf Ressourcen in Ihrem Konto erhalten haben.

Wir empfehlen die Verwendung der SourceAccount globalen Bedingungsschlüssel [aws:SourceArn](#) und [aws:](#) in Ressourcenrichtlinien, um die Berechtigungen einzuschränken, die der Ressource AWS Elemental MediaTailor einen anderen Dienst gewähren. Wenn Sie beide globalen Bedingungskontextschlüssel verwenden, müssen der `aws:SourceAccount`-Wert und das Konto im

`aws:SourceArn`-Wert dieselbe Konto-ID verwenden, wenn sie in derselben Richtlinienanweisung verwendet werden.

Der Wert von `aws:SourceArn` muss die Wiedergabekonfiguration sein, mit der CloudWatch Protokolle für Ihre Region und Ihr Konto veröffentlicht werden. Dies gilt jedoch nur, wenn Sie die [MediaTailorLogger](#)-Rolle verwenden, mit der CloudWatch Amazon-Protokolle in Ihrem Konto MediaTailor veröffentlicht werden können. Dies gilt nicht, wenn Sie eine [dienstbezogene Rolle](#) verwenden, um die CloudWatch Protokolle MediaTailor veröffentlichen zu lassen.

Der effektivste Weg, um sich vor dem Confused-Deputy-Problem zu schützen, ist die Verwendung des globalen Bedingungskontext-Schlüssels `aws:SourceArn` mit dem vollständigen ARN der Ressource. Wenn Sie den vollständigen ARN der Ressource nicht kennen oder wenn Sie mehrere Ressourcen angeben, verwenden Sie den globalen Bedingungskontext-Schlüssel `aws:SourceArn` mit Platzhaltern (*) für die unbekannten Teile des ARN. Beispiel, `arn:aws:service::123456789012:*`.

Das folgende Beispiel zeigt, wie Sie die Kontextschlüssel `aws:SourceArn` und die `aws:SourceAccount` globale Bedingung verwenden können, um das Problem des verwirrten Stellvertreters zu vermeiden.

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "mediatailor.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "ArnEquals": {
        "aws:SourceArn": "arn:aws:mediatailor:region:account_ID:playbackConfiguration/"
      },
      "StringEquals": {
        "aws:SourceAccount": "account_ID"
      }
    }
  }
}
```


Anmeldung und Überwachung MediaTailor

Dieser Abschnitt bietet eine Übersicht über die Optionen zur Protokollierung und Überwachung in AWS Elemental MediaTailor zu Sicherheitszwecken. Weitere Informationen zur Protokollierung und Überwachung MediaTailor finden Sie unter [Ressourcen überwachen und kennzeichnen AWS Elemental MediaTailor](#).

Die Überwachung ist ein wichtiger Bestandteil der Aufrechterhaltung der Zuverlässigkeit, Verfügbarkeit AWS Elemental MediaTailor und Leistung Ihrer AWS Lösungen. Sie sollten Überwachungsdaten aus allen Teilen Ihrer AWS Lösung sammeln, damit Sie einen etwaigen Ausfall an mehreren Stellen leichter debuggen können. AWS bietet verschiedene Tools zur Überwachung Ihrer MediaTailor Ressourcen und zur Reaktion auf potenzielle Vorfälle:

CloudWatch Amazon-Alarme

Mithilfe von CloudWatch Alarmen beobachten Sie eine einzelne Metrik über einen von Ihnen festgelegten Zeitraum. Wenn die Metrik einen bestimmten Schwellenwert überschreitet, wird eine Benachrichtigung an ein Amazon SNS SNS-Thema oder eine AWS Auto Scaling Scaling-Richtlinie gesendet. CloudWatch Alarme lösen keine Aktionen aus, da sie sich in einem bestimmten Status befinden. Der Status muss sich stattdessen geändert haben und für eine festgelegte Anzahl an Zeiträumen aufrechterhalten worden sein. Weitere Informationen finden Sie unter [the section called “Überwachung mit CloudWatch Metriken”](#).

AWS CloudTrail Logs

CloudTrail bietet eine Aufzeichnung der Aktionen, die von einem Benutzer, einer Rolle oder einem AWS Dienst in ausgeführt wurden AWS Elemental MediaTailor. Anhand der von gesammelten Informationen können Sie die Anfrage CloudTrail, an die die Anfrage gestellt wurde MediaTailor, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere Informationen ermitteln. Weitere Informationen finden Sie unter [AWS Elemental MediaTailor API-Aufrufe aufzeichnen](#).

AWS Trusted Advisor

Trusted Advisor stützt sich auf bewährte Verfahren, die wir bei der Betreuung von Hunderttausenden von AWS Kunden gelernt haben. Trusted Advisor untersucht Ihre AWS-Umgebung und gibt dann Empfehlungen, wenn Möglichkeiten bestehen, Geld zu sparen, die Systemverfügbarkeit und -leistung zu verbessern oder Sicherheitslücken zu schließen. Alle AWS -Kunden haben Zugriff auf fünf Trusted

Advisor-Prüfungen. Kunden mit dem „Business“- oder „Enterprise“-Support-Plan können alle Trusted Advisor -Überprüfungen anzeigen.

Weitere Informationen finden Sie unter [AWS Trusted Advisor](#).

Ressourcen überwachen und kennzeichnen AWS Elemental MediaTailor

Die Überwachung ist ein wichtiger Bestandteil der Aufrechterhaltung der Zuverlässigkeit, Verfügbarkeit und Leistung Ihrer anderen AWS Elemental MediaTailor Lösungen. AWS bietet die folgenden Überwachungstools, mit denen Sie beobachten MediaTailor, melden können, wenn etwas nicht stimmt, und gegebenenfalls automatische Maßnahmen ergreifen können:

- Amazon CloudWatch überwacht Ihre AWS Ressourcen und die Anwendungen, auf denen Sie laufen, AWS in Echtzeit. Sie können Kennzahlen erfassen und verfolgen, benutzerdefinierte Dashboards erstellen und Alarme festlegen, die Sie benachrichtigen oder Maßnahmen ergreifen, wenn eine bestimmte Metrik einen von Ihnen festgelegten Schwellenwert erreicht. Sie können beispielsweise die CPU-Auslastung oder andere Kennzahlen Ihrer EC2 Amazon-Instances CloudWatch verfolgen und bei Bedarf automatisch neue Instances starten. Weitere Informationen finden Sie im [CloudWatch Amazon-Benutzerhandbuch](#).
- Mit Amazon CloudWatch Logs können Sie Ihre Protokolldateien aus allen Interaktionen mit Ihrem Ad Decision Server (ADS) überwachen, speichern und darauf zugreifen. AWS Elemental MediaTailor sendet Protokolle für Anzeigenanfragen, Weiterleitungen und Antworten sowie für Berichtsanfragen und -antworten aus. Fehler von den ADS- und Ursprungsservern werden auch an Protokollgruppen in Amazon ausgegeben CloudWatch. Sie können Ihre Protokolldaten auch in einem sehr robusten Speicher archivieren. Allgemeine Informationen finden Sie im [Amazon CloudWatch Logs-Benutzerhandbuch](#). Informationen zu den ADS-Protokollen und wie Sie auf sie für Analysen über Amazon CloudWatch Logs Insights zugreifen können, finden Sie unter [AWS Elemental MediaTailor ADS-Protokollanalyse in Amazon CloudWatch Logs Insights](#).

Themen

- [AWS Elemental MediaTailor Logs anzeigen](#)
- [Überwachung AWS Elemental MediaTailor mit CloudWatch Amazon-Metriken](#)
- [AWS Elemental MediaTailor API-Aufrufe aufzeichnen](#)
- [Empfangen von AWS Elemental MediaTailor Channel Assembly-Benachrichtigungen](#)
- [Ressourcen taggen AWS Elemental MediaTailor](#)
- [Überwachung von AWS Mediendiensten mit Workflow-Monitor](#)

AWS Elemental MediaTailor Logs anzeigen

MediaTailor sendet Protokolle aus, die eine Vielzahl von Meilensteinen und Aktivitäten in Kanälen und Wiedergabekonfigurationen beschreiben. Sie können diese Protokolle verwenden, um sich einen Überblick über Ihren Arbeitsablauf zu verschaffen und Probleme mit dem Service zu beheben. In den folgenden Themen werden Protokolle und Protokollierungsoptionen beschrieben.

Themen

- [AWS Elemental MediaTailor Beschreibung und Ereignistypen der ADS-Protokolle](#)
- [AWS Elemental MediaTailor Beschreibung und Ereignistypen der Manifestprotokolle](#)
- [AWS Elemental MediaTailor Beschreibung und Ereignistypen der Transcodierungsprotokolle](#)
- [Verwenden von verkauften Protokollen zum Senden AWS Elemental MediaTailor von Protokollen](#)
- [AWS Elemental MediaTailor Logs direkt in Amazon CloudWatch Logs schreiben](#)
- [Steuerung des Volumens der Protokolle AWS Elemental MediaTailor](#)
- [Filterung von AWS Elemental MediaTailor Protokollen und Ereignissen](#)
- [AWS Elemental MediaTailor Debug-Protokolle generieren](#)

AWS Elemental MediaTailor Beschreibung und Ereignistypen der ADS-Protokolle

In den folgenden Abschnitten werden die Protokolle beschrieben, die MediaTailor zur Beschreibung von Ereignissen mit dem AD-Entscheidungsserver (ADS) ausgegeben werden. Dabei handelt es sich um AdDecisionServerInteractions Protokolle.

Themen

- [AdDecisionServerInteractions Veranstaltungen](#)
- [Beschreibung des ADS-Protokolls](#)
- [JSON-Schema für das ADS-Protokoll](#)

AdDecisionServerInteractions Veranstaltungen

Die folgenden Ereignisse werden bei MediaTailor Interaktionen mit dem Ad Decision Server (ADS) ausgelöst.

Protokoll	Beschreibung
AD_MARKER_FOUND	MediaTailor Ich habe eine Werbemarkierung im Manifest gefunden.
BEACON_FIRED	MediaTailor hat ein Tracking-Beacon abgefeuert.
EMPTY_VAST_RESPONSE	Die ADS hat eine leere VAST-Antwort zurückgegeben, die keine Werbung enthielt.
EMPTY_VMAP_RESPONSE	Das ADS hat eine leere VMAP-Antwort zurückgegeben.
ERROR_ADS_INVALID_RESPONSE	Das ADS hat einen Statuscode zurückgegeben, der nicht 200 ist.
ERROR_ADS_IO	MediaTailor Beim Versuch, mit dem ADS zu kommunizieren, ist ein Fehler aufgetreten.
ERROR_ADS_RESPONSE_PARSE	MediaTailor ist beim Analysieren der ADS-Antwort auf einen Fehler gestoßen.
ERROR_ADS_RESPONSE_UNKNOWN_ROOT_ELEMENT	Die ADS-Antwort enthält ein ungültiges Stammelement.
ERROR_ADS_TIMEOUT	Bei der MediaTailor Anfrage an das ADS wurde das Zeitlimit überschritten.
ERROR_DISALLOWED_HOST	Der ADS-Host ist nicht zulässig.
ERROR_FIRING_BEACON_FAILED	MediaTailor konnte das Tracking-Beacon nicht abfeuern.
ERROR_PERSONALIZATION_DISABLED	Das Einfügen von Anzeigen ist für diese Sitzung deaktiviert.
ERROR_UNKNOWN	MediaTailor ist bei der ADS-Anfrage auf einen unbekannten Fehler gestoßen.

Protokoll	Beschreibung
ERROR_UNKNOWN_HOST	Der ADS-Host ist unbekannt.
ERROR_VAST_INVALID_MEDIA_FILE	Das Ad VAST-Objekt enthält ein ungültiges oder MediaFile fehlendes Element.
ERROR_VAST_INVALID_VAST_AD_TAG_URI	Die VAST-Antwort enthält ein ungültige sVASTAdTagURI .
ERROR_VAST_MISSING_CREATIVES	Das VAST Ad enthält null oder mehrere Creatives Elemente. Genau ein Creatives Element ist erforderlich.
ERROR_VAST_MISSING_IMPRESSION	Das VAST Ad enthält keine Impression Elemente. Es ist mindestens ein Impression Element erforderlich.
ERROR_VAST_MISSING_MEDIAFILES	Das VAST Ad enthält null oder mehrere MediaFiles Elemente. Genau ein MediaFiles Element ist erforderlich.
ERROR_VAST_MISSING_OVERLAYS	MediaTailor hat keine nichtlinearen Werbeanzeigen vom Ad-Server erhalten.
ERROR_VAST_MULTIPLE_LINEAR	Das VAST Ad enthält mehrere Linear Elemente.
ERROR_VAST_MULTIPLE_TRACKING_EVENTS	Das VAST Ad enthält mehrere TrackingEvents Elemente.
ERROR_VAST_REDIRECT_EMPTY_RESPONSE	Die VAST-Umleitungsanfrage gab eine leere Antwort zurück.
ERROR_VAST_REDIRECT_FAILED	Bei der VAST-Umleitungsanfrage ist ein Fehler aufgetreten.
ERROR_VAST_REDIRECT_MULTIPLE_VAST	Bei der VAST-Weiterleitungsanfrage wurden mehrere Anzeigen zurückgegeben.

Protokoll	Beschreibung
FILLED_AVAIL	MediaTailor Der Vorrat wurde erfolgreich ausgefüllt.
FILLED_OVERLAY_AVAIL	MediaTailor Das Overlay-Angebot wurde erfolgreich ausgefüllt.
INTERSTITIAL_VOD_FAILURE	Bei der ADS-Anfrage oder -Antwort ist beim Ausfüllen der Interstitial-Avails für die VOD-Wiedergabeliste ein Problem aufgetreten. Es werden keine Anzeigen eingefügt.
INTERSTITIAL_VOD_SUCCESS	MediaTailor Interstitielle Avails für die VOD-Playlist wurden erfolgreich ausgefüllt.
MAKING_ADS_REQUEST	MediaTailor fordert Werbung von der ADS an.
MODIFIED_TARGET_URL	MediaTailor hat die ausgehende Ziel-URL geändert.
NON_AD_MARKER_FOUND	MediaTailor Ich habe im Manifest eine Werbemarkierung gefunden, die nicht bearbeitet werden kann.
RAW_ADS_RESPONSE	MediaTailor hat eine unformatierte ADS-Antwort erhalten.
REDIRECTED_VAST_RESPONSE	MediaTailor erhielt eine VAST-Antwort, nachdem ich der VAST-Weiterleitung gefolgt war.
VAST_REDIRECT	Die VAST-Anzeigenantwort enthält eine Weiterleitung.
VAST_RESPONSE	MediaTailor hat eine VAST-Antwort erhalten.
VOD_TIME_BASED_AVAIL_PLAN_SUCCESS	MediaTailor erfolgreich einen zeitbasierten Verfügbarkeitsplan für die VOD-Vorlage erstellt.

Protokoll	Beschreibung
VOD_TIME_BASED_AVAIL_PLAN_VAST_RESPONSE_FOR_OFFSET	MediaTailor erstellt einen zeitbasierten Nutzungsplan für die VOD-Vorlage. MediaTailor hat eine VAST-Antwort für den Zeitversatz erhalten.
VOD_TIME_BASED_AVAIL_PLAN_WARNING_NO_ADVERTISEMENTS	Bei der ADS-Anfrage oder -Antwort ist beim Erstellen eines zeitbasierten Verfügbarkeitsplans für die VOD-Vorlage ein Problem aufgetreten. Es werden keine Anzeigen eingefügt.
WARNING_NO_ADVERTISEMENTS	MediaTailor ist beim Ausfüllen des Formulars auf ein Problem gestoßen. Es wurden keine Anzeigen eingefügt.
WARNING_URL_VARIABLE_SUBSTITUTION_FAILED	MediaTailor kann dynamische Variablen in der ADS-URL nicht ersetzen. Überprüfen Sie die URL-Konfiguration.
WARNING_VPAID_AD_DROPPED	Eine VPAID-Anzeige wurde aufgrund eines fehlenden Slate gelöscht, oder die Sitzung verwendet serverseitige Berichte.

Beschreibung des ADS-Protokolls

In diesem Abschnitt werden die Struktur und der Inhalt der ADS-Protokollbeschreibung beschrieben. Um sich selbst in einem JSON-Editor zu beschäftigen, verwenden Sie die Auflistung unter [the section called “JSON-Schema für das ADS-Protokoll”](#).

Jedes Ereignis im ADS-Protokoll enthält die Standardfelder, die von CloudWatch Logs generiert werden. Weitere Informationen finden Sie unter [Analysieren von Protokolldaten mit CloudWatch Logs Insights](#).

ADS protokolliert Eigenschaften

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle beschrieben.

Eigenschaft	Typ	Erforderlich	Beschreibung
<code>adsRequestUrl</code>	Zeichenfolge	false	Die vollständige URL der ADS-Anfrage von MediaTailor.
<code>avail</code>	Objekt des Typs avail	false	Informationen zu einem Angebot, das voller MediaTailor Werbung ist. Derzeit ist dies für den FILLED_AVAIL Ereignistyp der Plan, der erstellt wird, MediaTailor wenn er zum ersten Mal auf den Erfolg stößt. Wie der avail letztendlich gefüllt wird, kann sich von diesem Plan unterscheiden, je nachdem, wie die Inhalte abgespielt werden.
<code>awsAccountId</code>	Zeichenfolge	true	Die AWS-Konto-ID für die MediaTailor Konfiguration, die für die Sitzung verwendet wurde.
<code>customerId</code>	Zeichenfolge	true	Die gehashte Version der AWS-Konto-ID, die Sie verwenden können, um mehrere Protokolleinträge zu korrelieren.

Eigenschaft	Typ	Erforderlich	Beschreibung
eventDescription	Zeichenfolge	true	Eine kurze Beschreibung des Ereignisses, das diese Protokollnachricht ausgelöst hat, bereitgestellt vom MediaTailor Service. Standardmäßig ist dieses Feld leer. Beispiel: Got VAST response.
eventTimestamp	Zeichenfolge	true	Das Datum und die Uhrzeit des Ereignisses.
eventType	Zeichenfolge	true	Der Code für das Ereignis, das diese Protokollnachricht ausgelöst hat. Beispiel: VAST_RESPONSE .
originId	Zeichenfolge	true	Der Konfigurationsname aus der MediaTailor Konfiguration. Dies unterscheidet sich von der Quelle der Videoinhalte, die ebenfalls Teil der Konfiguration ist.

Eigenschaft	Typ	Erforderlich	Beschreibung
<code>requestHeaders</code>	Array vom Typ requestheaders	false	Die Header, die in der ADS-Anfrage MediaTailor enthalten waren. In der Regel enthalten die Protokolle diese, wenn eine Anforderung an den ADS fehlschlägt, um die Fehlerbehebung zu unterstützen.
<code>requestId</code>	Zeichenfolge	true	Die MediaTailor Anforderungs-ID, mit der Sie mehrere Protokolleinträge für dieselbe Anfrage korrelieren können.
<code>sessionId</code>	Zeichenfolge	true	Die eindeutige numerische Kennung, die MediaTailor der Spiellersitzung zugewiesen wurde. Alle Anforderungen, die ein Spieler für eine Sitzung stellt, haben dieselbe Sitzungs-ID. Beispiel: e039fd39-09f0-46b2-aca9-9871cc116cde .
<code>sessionType</code>	Zeichenfolge (zulässige Werte: [DASH, HLS])	true	Der Stream-Typ des Spielers.

Eigenschaft	Typ	Erforderlich	Beschreibung
vastAd	Objekt des Typs vastAd	false	Informationen zu einer einzelnen Anzeige, die aus der VAST-Antwort analysiert wurde.
vastResponse	Objekt des Typs vastResponse	false	Informationen über die VAST-Antwort, die vom ADS MediaTailor empfangen wurde.
vodCreativeOffsets	Objekt des Typs vodCreativeOffsets	false	Eine Übersicht, die die Zeitversätze im Manifest angibt, bei denen Avails auf der Grundlage der VMAP-Antwort eingefügt MediaTailor werden.
vodVastResponseTimeOffset	Zahl	false	Der VMAP-spezifische Zeitversatz zum Einfügen der VOD-Anzeige.

adContent

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle adContent beschrieben.

Eigenschaften der ADS-Protokolle adContent

Eigenschaft	Typ	Erforderlich	Beschreibung
adPlaylistUri	Objekt des Typs adPlaylistUri	false	Die Zuordnung vom Ursprungsmanifest für eine Variante zum Anzeigenmanifest

Eigenschaft	Typ	Erforderlich	Beschreibung
			für die Variante. Für DASH enthält dies einen einzelnen Eintrag, da alle Varianten in einem einzigen DASH-Manifest dargestellt werden.

adPlaylistUri

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle beschrieben adPlaylistUri.

adPlaylistUri Eigenschaften von ADS-Protokollen

Eigenschaft	Typ	Erforderlich	Beschreibung
<any string>	Zeichenfolge	false	Die URL des Anzeigenmanifests für die spezifische Variante.

avail

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle avail beschrieben.

Eigenschaften der ADS-Protokolle avail

Eigenschaft	Typ	Erforderlich	Beschreibung
availId	Zeichenfolge	true	Die eindeutige Kennung für diese avail-Protokoll. Bei HLS ist dies die Mediensequenznummer, bei der der avail beginnt. Für

Eigenschaft	Typ	Erforderlich	Beschreibung
			DASH ist dies die Punkt-ID.
creativeAds	Array vom Typ creativeAd	true	Die Anzeigen, die in den Avail MediaTailor eingefügt wurden.
fillRate	Zahl	true	Die Rate, mit der die Anzeigen die avail-Dauer erfüllen, von 0,0 (für 0 %) bis 1,0 (für 100 %).
filledDuration	Zahl	true	Die Summe der Dauer aller Anzeigen, die in avail eingefügt wurden.
numAds	Zahl	true	Die Anzahl der Anzeigen, die in den Avail MediaTailor eingefügt wurden.
originAvailabilityDuration	Zahl	true	Die Dauer des avails, wie im Inhalts-Stream vom Ursprung angegeben (CUE_OUT oder SCTE).

Eigenschaft	Typ	Erforderlich	Beschreibung
skippedAds	Array vom Typ skippedAd	false	Die Anzeigen, die aus Gründen wie TRANSCODE_IN_PROGRESS und MediaTailor TRANSCODE_ERROR nicht eingefügt wurden.
slateAd	Objekt des Typs slateAd	true	Informationen über die Slate-Anzeige, mit MediaTailor der alle nicht ausgefüllten Segmente gefüllt werden, sind verfügbar.

creativeAd

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle creativeAd beschrieben.

Eigenschaften der ADS-Protokolle creativeAd

Eigenschaft	Typ	Erforderlich	Beschreibung
adContent	Objekt des Typs adContent	true	Informationen über den Inhalt der eingefügten Anzeige.
creativeUniqueId	Zeichenfolge	true	Die eindeutige Kennung für die Anzeige, die als Schlüssel für die Transcodierung verwendet wird. Dies ist das ID-Feld für das

Eigenschaft	Typ	Erforderlich	Beschreibung
			Werbemittel in der VAST-Antwort, sofern verfügbar. Andernfalls handelt es sich um die Mezzanine-URL der Anzeige.
trackingEvents	Objekt des Typs trackingEvents	true	Das Tracking-Beacon URLs für die verschiedenen Tracking-Ereignisse für die Anzeige. Die Schlüssel sind die Namen der Ereignisse, und die Werte sind eine Liste von Beacons URLs.
transcodeAdDuration	Zahl	true	Die Dauer der Anzeige, berechnet aus der transcodierten Komponente.
uri	Zeichenfolge	true	Die URL der Mezzanine-Version der Anzeige, bei der es sich um die Eingabe für den Transcoder handelt.
vastDuration	Zahl	true	Die Dauer der Anzeige, wie aus der VAST-Antwort analysiert.

requestheaders

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle requestheaders beschrieben.

Eigenschaften der ADS-Protokolle requestheaders

Eigenschaft	Typ	Erforderlich	Beschreibung
name	Zeichenfolge	true	Der Name des hinzuzufügenden Headers.
value	Zeichenfolge	true	Der Wert des Headers

skippedAd

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle skippedAd beschrieben.

Eigenschaften der ADS-Protokolle skippedAd

Eigenschaft	Typ	Erforderlich	Beschreibung
adMezzanineUrl	Zeichenfolge	true	Die Mezzanine-URL der übersprungenen Anzeige.
creativeUniqueId	Zeichenfolge	true	Die eindeutige Kennung für die Anzeige, die als Schlüssel für die Transcodierung verwendet wird. Dies ist das ID-Feld für das Werbemittel in der VAST-Antwort, sofern verfügbar. Andernfalls handelt es sich um die Mezzanine-URL der Anzeige.

Eigenschaft	Typ	Erforderlich	Beschreibung
skippedReason	Zeichenfolge	true	Der Code, der angibt, warum die Anzeige nicht eingefügt wurde. Beispiel: TRANSCODE_IN_PROGRESS .
transcodeAdDuration	Zahl	false	Die Dauer der Anzeige, berechnet aus der transcodierten Komponente.
vastDuration	Zahl	true	Die Dauer der Anzeige, wie aus der VAST-Antwort analysiert.

slateAd

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle slateAd beschrieben.

Eigenschaften der ADS-Protokolle slateAd

Eigenschaft	Typ	Erforderlich	Beschreibung
adContent	Objekt des Typs adContent	true	Informationen über den Inhalt der eingefügten Anzeige.
creativeUniqueId	Zeichenfolge	true	Die eindeutige Kennung für die Anzeige, die als Schlüssel für die Transcodierung verwendet wird. Dies ist das ID-Feld für das Werbemittel in der

Eigenschaft	Typ	Erforderlich	Beschreibung
			VAST-Antwort, sofern verfügbar. Andernfalls handelt es sich um die Mezzanine-URL der Anzeige.
<code>transcodeAdDuration</code>	Zahl	true	Die Dauer der Anzeige, berechnet aus der transcodierten Komponente.
<code>uri</code>	Zeichenfolge	true	Die URL der Mezzanine-Version der Anzeige, bei der es sich um die Eingabe für den Transcoder handelt.

trackingEvents

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle trackingEvents beschrieben.

Eigenschaften der ADS-Protokolle trackingEvents

Eigenschaft	Typ	Erforderlich	Beschreibung
<code><any string></code>	Array vom Typ Zeichenfolge	false	Die Liste der Beacons URLs für das angegebene Tracking-Ereignis (Impression, Complete usw.)

vastAd

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle vastAd beschrieben.

Eigenschaften der ADS-Protokolle vastAd

Eigenschaft	Typ	Erforderlich	Beschreibung
adSystem	Zeichenfolge	true	Der Wert des AdSystem-Tags in der VAST-Antwort.
adTitle	Zeichenfolge	true	Die Mediendateien, die für die Anzeige in der VAST-Antwort verfügbar sind.
creativeAdId	Zeichenfolge	true	Der Wert des adId-Attributs des Creative-Tags in der VAST-Antwort.
creativeId	Zeichenfolge	true	Der Wert des id-Attributs des Creative-Tags in der VAST-Antwort.
duration	Zahl	true	Die ungefähre Dauer der Anzeige, basierend auf dem duration-Tag im linear-Element der VAST-Antwort.
trackingEvents	Objekt des Typs trackingEvents	true	Das Tracking-Beacon URLs für die verschiedenen Tracking-Ereignisse für die Anzeige. Die Schlüssel sind die Namen der Ereignisse, und die Werte

Eigenschaft	Typ	Erforderlich	Beschreibung
			sind eine Liste von Beacons URLs.
vastAdId	Zeichenfolge	true	Der Wert des id-Attributs des Ad-Tags in der VAST-Antwort
vastAdTagUri	Zeichenfolge	false	Die VMAP-spezifische Umleitungs-URI für eine Anzeige.
vastMediaFiles	Array vom Typ vastMediaFile	true	Die Liste der verfügbaren Mediendateien für die Anzeige in der VAST-Antwort.

vastMediaFile

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle beschrieben vastMediaFile.

vastMediaFile Eigenschaften von ADS-Protokollen

Eigenschaft	Typ	Erforderlich	Beschreibung
apiFramework	Zeichenfolge	true	Das API-Framework, das für die Verwaltung der Mediendatei erforderlich ist. Beispiel: VPAID.
bitrate	Zahl	true	Die Bitrate der Mediendatei.
delivery	Zeichenfolge	true	Das Protokoll, das für die Mediendatei verwendet wird, ist

Eigenschaft	Typ	Erforderlich	Beschreibung
			auf progressiv oder Streaming festgelegt.
height	Zahl	true	Die Pixelhöhe der Mediendatei.
id	Zeichenfolge	true	Der Wert des id-Attributs des MediaFile -Tags.
type	Zeichenfolge	true	Der MIME-Typ der Mediendatei aus dem type-Attribut des MediaFile -Tags.
uri	Zeichenfolge	true	Die URL der Mezzanine-Version der Anzeige, bei der es sich um die Eingabe für den Transcoder handelt.
width	Zahl	true	Die Pixelbreite der Mediendatei.

vastResponse

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle vastResponse beschrieben.

Eigenschaften der ADS-Protokolle vastResponse

Eigenschaft	Typ	Erforderlich	Beschreibung
errors	Array vom Typ Zeichenfolge	true	Der Fehler URLs wurde anhand der Error Tags in

Eigenschaft	Typ	Erforderlich	Beschreibung
			der VAST-Antwort analysiert.
<code>vastAds</code>	Array vom Typ vastAd	true	Die Anzeigen, die aus der VAST-Antwort analysiert wurden.
<code>version</code>	Zeichenfolge	true	Die VAST-Spezifikationsversion, die aus dem <code>version</code> -Attribut des VAST-Tags in der Antwort analysiert wird.

vodCreativeOffsets

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle beschrieben vodCreativeOffsets.

vodCreativeOffsets Eigenschaften von ADS-Protokollen

Eigenschaft	Typ	Erforderlich	Beschreibung
<code><any string></code>	Array vom Typ vodCreativeOffset	false	Eine Zuordnung von einem Zeitversatz im Manifest zu einer Liste von Anzeigen, die zu diesem Zeitpunkt eingefügt werden sollen.

vodCreativeOffset

In diesem Abschnitt werden die Eigenschaften der ADS-Protokolle beschrieben vodCreativeOffset.

vodCreativeOffset Eigenschaften von ADS-Protokollen

Eigenschaft	Typ	Erforderlich	Beschreibung
adContent	Objekt des Typs adContent	true	Informationen über den Inhalt der eingefügten Anzeige.
creativeUniqueId	Zeichenfolge	true	Die eindeutige Kennung für die Anzeige, die als Schlüssel für die Transcodierung verwendet wird. Dies ist das ID-Feld für das Werbemittel in der VAST-Antwort, sofern verfügbar. Andernfalls handelt es sich um die Mezzanine-URL der Anzeige.
trackingEvents	Objekt des Typs trackingEvents	true	Das Tracking-Beacon URLs für die verschiedenen Tracking-Ereignisse für die Anzeige. Die Schlüssel sind die Namen der Ereignisse, und die Werte sind eine Liste von Beacons URLs.
transcodedAdDuration	Zahl	true	Die Dauer der Anzeige, berechnet aus der transcodierten Komponente.

Eigenschaft	Typ	Erforderlich	Beschreibung
<code>uri</code>	Zeichenfolge	true	Die URL der Mezzanine-Version der Anzeige, bei der es sich um die Eingabe für den Transcoder handelt.
<code>vastDuration</code>	Zahl	true	Die Dauer der Anzeige, wie aus der VAST-Antwort analysiert.

JSON-Schema für das ADS-Protokoll

Im Folgenden ist das JSON-Schema für das AWS Elemental MediaTailor ADS-Protokoll aufgeführt.

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "http://amazon.com/elemental/midas/mms/adsLogSchema.json",
  "type": "object",
  "title": "AWS Elemental MediaTailor ADS Log JSON Schema",
  "required": [
    "eventType",
    "eventTimestamp",
    "requestId",
    "sessionType",
    "eventDescription",
    "awsAccountId",
    "customerId",
    "originId",
    "sessionId"
  ],
  "additionalProperties": false,
  "properties": {
    "eventType": {
      "$id": "#/properties/eventType",
      "type": "string",
```

```

    "description": "The code for the event that triggered this log message. Example:
<code>VAST_RESPONSE</code>.",
    "examples": [
        "FILLED_AVAIL"
    ]
},
"eventTimestamp": {
    "$id": "#/properties/eventTimestamp",
    "type": "string",
    "description": "The date and time of the event.",
    "examples": [
        "1970-01-01T00:00:00Z"
    ],
    "format": "date-time"
},
"requestId": {
    "$id": "#/properties/requestId",
    "type": "string",
    "description": "The MediaTailor request ID, which you can use to correlate
multiple log entries for the same request.",
    "examples": [
        "c7c7ae8c-a61e-44e0-8efd-7723995337a1"
    ],
    "pattern": "^(.*)$"
},
"sessionType": {
    "$id": "#/properties/sessionType",
    "type": "string",
    "enum": [
        "HLS",
        "DASH"
    ],
    "description": "The player's stream type."
},
"eventDescription": {
    "$id": "#/properties/eventDescription",
    "type": "string",
    "description": "A short description of the event that triggered this log message,
provided by the MediaTailor service. By default, this is empty. Example: <code>Got
VAST response</code>.",
    "default": "",
    "examples": [
        "Got VAST response"
    ],

```

```

    "pattern": "^(.*)$"
  },
  "awsAccountId": {
    "$id": "#/properties/awsAccountId",
    "type": "string",
    "description": "The AWS account ID for the MediaTailor configuration that was
used for the session."
  },
  "customerId": {
    "$id": "#/properties/customerId",
    "type": "string",
    "description": "The hashed version of the AWS account ID, which you can use to
correlate multiple log entries.",
    "pattern": "^(.*)$"
  },
  "originId": {
    "$id": "#/properties/originId",
    "type": "string",
    "description": "The configuration name from the MediaTailor configuration. This
is different from the video content source, which is also part of the configuration.",
    "examples": [
      "external-canary-dash-serverside-reporting-onebox"
    ],
    "pattern": "^(.*)$"
  },
  "sessionId": {
    "$id": "#/properties/sessionId",
    "type": "string",
    "description": "The unique numeric identifier that MediaTailor assigned to the
player session. All requests that a player makes for a session have the same session
ID. Example: <code>e039fd39-09f0-46b2-aca9-9871cc116cde</code>.",
    "examples": [
      "120b9873-c007-40c8-b3db-0f1bd194970b"
    ],
    "pattern": "^(.*)$"
  },
  "avail": {
    "$id": "#/properties/avail",
    "type": "object",
    "title": "avail",
    "description": "Information about an avail that MediaTailor fills with ads.
Currently, for the <code>FILLED_AVAIL</code> event type, this is the plan created by
MediaTailor when it first encounters the avail. How the avail is eventually filled may
vary from this plan, depending on how the content plays out. ",

```

```

    "required": [
      "creativeAds",
      "originAvailDuration",
      "filledDuration",
      "fillRate",
      "driftMillisecondsAtAvailStart",
      "numAds",
      "slateAd",
      "availId"
    ],
    "additionalProperties": false,
    "properties": {
      "originAvailDuration": {
        "$id": "#/properties/avail/originAvailDuration",
        "type": "number",
        "description": "The duration of the avail as specified in the content stream
from the origin (<code>CUE_OUT</code> or <code>SCTE</code>).",
      },
      "filledDuration": {
        "$id": "#/properties/avail/filledDuration",
        "type": "number",
        "description": "The sum of the durations of all the ads inserted into the
avail.",
      },
      "fillRate": {
        "$id": "#/properties/avail/fillRate",
        "type": "number",
        "description": "The rate at which the ads fill the avail duration, from 0.0
(for 0%) to 1.0 (for 100%).",
      },
      "driftMillisecondsAtAvailStart": {
        "$id": "#/properties/avail/driftMillisecondsAtAvailStart",
        "type": "number",
        "description": "The cumulative drift at the beginning of this avail. A
positive value implies that we are moving away from the live edge, a negative value
implies that we are moving towards the live edge.",
      },
      "creativeAds": {
        "$id": "#/properties/avail/creativeAds",
        "type": "array",
        "description": "The ads that MediaTailor inserted into the avail.",
        "items": {
          "type": "object",
          "title": "creativeAd",

```

```

    "description": "Information about a single inserted ad.",
    "required": [
      "uri",
      "creativeUniqueId",
      "adSystem",
      "adContent",
      "trackingEvents",
      "vastDuration",
      "transcodedAdDuration"
    ],
    "additionalProperties": false,
    "properties": {
      "uri": { "$ref": "#/definitions/adMezzanineUri" },
      "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
      "adSystem": { "$ref": "#/definitions/adSystem" },
      "adContent": { "$ref": "#/definitions/adContent" },
      "trackingEvents": { "$ref": "#/definitions/trackingEvents" },
      "vastDuration": { "$ref": "#/definitions/vastDuration" },
      "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" }
    }
  },
  "numAds": {
    "$id": "#/properties/avail/numAds",
    "type": "number",
    "description": "The number of ads that MediaTailor inserted into the avail."
  },
  "slateAd": {
    "$id": "#/properties/avail/slateAd",
    "type": ["object", "null"],
    "title": "slateAd",
    "description": "Information about the slate ad, which MediaTailor uses to
fill any unfilled segments in the avail.",
    "additionalProperties": false,
    "required": [
      "uri",
      "creativeUniqueId",
      "adContent",
      "transcodedAdDuration"
    ],
    "properties": {
      "uri": { "$ref": "#/definitions/adMezzanineUri" },
      "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
      "adContent": { "$ref": "#/definitions/adContent" },

```

```

    "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" }
  },
  "availId": {
    "$id": "#/properties/avail/availId",
    "type": "string",
    "description": "The unique identifier for this avail. For HLS, this is the
media sequence number where the avail begins. For DASH, this is the period ID."
  },
  "skippedAds": {
    "$id": "#/properties/avail/skippedAds",
    "type": "array",
    "description": "The ads that MediaTailor didn't insert, for reasons like
<code>TRANSCODE_IN_PROGRESS</code> and <code>TRANSCODE_ERROR</code>.",
    "items": {
      "type": "object",
      "title": "skippedAd",
      "description": "Information about a single skipped ad.",
      "required": [
        "creativeUniqueId",
        "adMezzanineUrl",
        "skippedReason",
        "vastDuration"
      ],
      "additionalProperties": false,
      "properties": {
        "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
        "adMezzanineUrl": {
          "type": "string",
          "description": "The mezzanine URL of the skipped ad."
        },
        "skippedReason": {
          "type": "string",
          "description": "The code that indicates why the ad wasn't inserted.
Example: <code>TRANSCODE_IN_PROGRESS</code>."
        },
        "vastDuration": { "$ref": "#/definitions/vastDuration" },
        "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" },
        "targetVariant": {
          "type": "object",
          "title": "targetVariant",
          "description": "The target variant of the source content. This key is
present when an ad wasn't inserted due to the source content containing a variant that
could not match to any variants present in this ad.",

```

```
"required": [
  "mediaProtocol",
  "mediaType",
  "bitrate",
  "mediaResolution",
  "codecs"
],
"additionalProperties": false,
"properties": {
  "mediaProtocol": {
    "type": "string",
    "description": "The media protocol of this variant, such as HLS.",
    "enum": [
      "HLS",
      "DASH"
    ]
  },
  "mediaType": {
    "type": "array",
    "description": "The media type of this variant, such as VIDEO.",
    "items": {
      "type": "string",
      "enum": [
        "VIDEO",
        "AUDIO",
        "SUBTITLES",
        "TRICK_PLAY"
      ]
    },
    "description": "Media type, such as VIDEO."
  },
  "bitrate": {
    "$ref": "#/definitions/bitrate"
  },
  "mediaResolution": {
    "type": "object",
    "title": "mediaResolution",
    "description": "The media resolution of this variant.",
    "required": [
      "width",
      "height"
    ],
    "additionalProperties": false,
    "properties": {
```

```

        "width": {
            "$ref": "#/definitions/width"
        },
        "height": {
            "$ref": "#/definitions/height"
        }
    },
    "codecs": {
        "type": "array",
        "description": "The codecs of this variant.",
        "items": {
            "type": "string",
            "description": "Codec, such as avc1."
        }
    }
}

},
"adBreakTrackingEvents": {
    "$id": "#properties/avail/adBreakTrackingEvents",
    "type": "object",
    "title": "adBreakTrackingEvents",
    "description": "VMAP/ad break tracking events and corresponding URL",
    "properties": {
        "items": {
            "$ref": "#/definitions/adBreakTrackingEvents"
        }
    }
}

},
"vastResponse": {
    "$id": "#/properties/vastResponse",
    "type": "object",
    "title": "vastResponse",
    "description": "Information about the VAST response that MediaTailor received from the ADS.",
    "required": [
        "version",
        "vastAds",

```



```

    "errors",
    "nonLinearAdsList"
  ],
  "additionalProperties": false,
  "properties": {
    "version": {
      "$id": "#/properties/vastResponse/version",
      "type": "string",
      "description": "The VAST specification version, parsed from the
<code>version</code> attribute of the <code>VAST</code> tag in the response.",
      "examples": [
        "3.0"
      ],
      "pattern": "^(.*)$"
    },
    "vastAds": {
      "$id": "#/properties/vastResponse/vastAds",
      "type": "array",
      "description": "The ads parsed from the VAST response.",
      "items": {
        "$ref": "#/definitions/vastAd"
      }
    },
    "errors": {
      "$id": "#/properties/vastResponse/errors",
      "type": "array",
      "description": "The error URLs parsed from the <code>Error</code> tags in the
VAST response.",
      "items": {
        "type": "string",
        "description": "A single error URL."
      }
    },
    "nonLinearAdsList": {
      "$id": "#/properties/vastResponse/nonLinearAds",
      "type": "array",
      "description": "A list of NonLinearAds as they are read from the VAST
response.",
      "items": {
        "$ref": "#/definitions/nonLinearAds"
      }
    }
  }
},

```

```

"vastAd": {
  "$ref": "#/definitions/vastAd"
},

"vodVastResponseTimeOffset": {
  "$id": "#/properties/vodVastResponseTimeOffset",
  "type": "number",
  "description": "The VMAP specific time offset for VOD ad insertion.",
  "examples": [
    5.0
  ]
},

"vodCreativeOffsets": {
  "$id": "#/properties/vodCreativeOffsets",
  "type": "object",
  "title": "vodCreativeOffsets",
  "description": "A map that indicates the time offsets in the manifest where
MediaTailor will insert avails, based on the VMAP response.",
  "additionalProperties": {
    "type": "array",
    "$id": "#/properties/vodCreativeOffsets/entry",
    "description": "A mapping from a time offset in the manifest to a list of ads
to insert at this time.",
    "items": {
      "type": "object",
      "$id": "#/properties/vodCreativeOffsets/entry/items",
      "title": "vodCreativeOffset",
      "description": "The list of ads to insert at the specified time offset.",
      "additionalProperties": false,
      "required": [
        "uri",
        "creativeUniqueId",
        "vastDuration",
        "transcodedAdDuration",
        "adContent",
        "trackingEvents"
      ],
      "properties": {
        "uri": { "$ref": "#/definitions/adMezzanineUri" },
        "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
        "vastDuration": { "$ref": "#/definitions/vastDuration" },
        "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" },

```

```

        "adContent": { "$ref": "#/definitions/adContent" },
        "trackingEvents": { "$ref": "#/definitions/trackingEvents" }
    }
}
},

"adsRequestUrl": {
    "$id": "#/properties/adsRequestUrl",
    "type": "string",
    "description": "The full URL of the ADS request made by MediaTailor."
},
"adMarkers": {
    "$id": "#/properties/adMarkers",
    "type": "string",
    "description": "Found Ad Marker in the Manifest."
},
"segmentationUpid": {
    "$id": "#/properties/segmentationUpid",
    "type": "string",
    "description": "Value of segmentation upid parsed from ad markers in manifest."
},
"segmentationTypeId": {
    "$id": "#/properties/segmentationTypeId",
    "type": "integer",
    "description": "Value of segmentation typeId parsed from ad markers in manifest."
},
"requestHeaders": {
    "$id": "#/properties/requestHeaders",
    "type": "array",
    "description": "The headers that MediaTailor included with the ADS request.
Typically, the logs include these when a request to the ADS fails, to help with
troubleshooting.",
    "items": {
        "type": "object",
        "title": "requestheaders",
        "description": "The name and value for a single header included in the ADS
request.",
        "required": [
            "name",
            "value"
        ],
        "additionalProperties": false,
        "properties": {

```

```

    "name": {
      "type": "string",
      "description": "The name of the header."
    },
    "value": {
      "type": "string",
      "description": "The value of the header."
    }
  }
},

"originalTargetUrl": {
  "$id": "#/properties/originalTargetUrl",
  "type": "string",
  "description": "The old URL to which MediaTailor was going to make a request."
},
"updatedTargetUrl": {
  "$id": "#/properties/updatedTargetUrl",
  "type": "string",
  "description": "The new URL to which MediaTailor is making a request."
},

"rawAdsResponse": {
  "$id": "#/properties/rawAdsResponse",
  "type": "string",
  "description": "Paginated ADS response as it's exactly returned to MediaTailor."
},
"rawAdsResponseIndex": {
  "$id": "#/properties/rawAdsResponseIndex",
  "type": "integer",
  "description": "Integer value denoting this rawAdsResponse's index into the
full ADS response. This value is used to order the paginated messages for this ADS
response."
}
},

"__COMMENT_oneOf": "The oneOf section defines subtypes for our events. Subtypes can
have different rules, including which fields are required. For more information, see
https://json-schema.org/understanding-json-schema/reference/combining.html#oneof ",

"oneOf": [
  { "$ref": "#/definitions/eventAdMarkersFound" },
  { "$ref": "#/definitions/eventNonAdMarkerFound" },

```

```

{ "$ref": "#/definitions/eventMakingAdsRequest" },
{ "$ref": "#/definitions/eventModifiedTargetUrl" },
{ "$ref": "#/definitions/eventRawAdsResponse" },
{ "$ref": "#/definitions/eventVastResponse" },
{ "$ref": "#/definitions/eventFilledAvail" },
{ "$ref": "#/definitions/eventFilledOverlayAvail" },
{ "$ref": "#/definitions/eventErrorFiringBeaconFailed" },
{ "$ref": "#/definitions/eventWarningNoAdvertisements" },
{ "$ref": "#/definitions/eventUnknownHost" },
{ "$ref": "#/definitions/eventErrorAdsTimeout" },
{ "$ref": "#/definitions/eventErrorVastMissingOverlays" },
{ "$ref": "#/definitions/eventPlannedAvail" },
{ "$ref": "#/definitions/eventEmptyVastResponse" },
{ "$ref": "#/definitions/eventEmptyVmapResponse" },
{ "$ref": "#/definitions/eventErrorUnknown" },
{ "$ref": "#/definitions/eventVastRedirect" },
{ "$ref": "#/definitions/eventRedirectedVastResponse" },
{ "$ref": "#/definitions/eventErrorAdsMissingImpression" },
{ "$ref": "#/definitions/eventErrorAdsResponseParse" },
{ "$ref": "#/definitions/eventErrorAdsInvalidResponse" },
{ "$ref": "#/definitions/eventErrorDisallowedHost" },
{ "$ref": "#/definitions/eventPersonalizationDisabled" },
{ "$ref": "#/definitions/eventWarningDynamicVariableSubFailed" },
{ "$ref": "#/definitions/eventVodTimeBasedAvailPlanVastResponseForOffset" },
{ "$ref": "#/definitions/eventVodTimeBasedAvailPlanSuccess" }
],

"definitions": {
  "eventAdMarkersFound": {
    "$id": "#/definitions/eventAdMarkersFound",
    "required": [
      "eventType",
      "adMarkers"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "AD_MARKER_FOUND"
      }
    }
  },
  "eventNonAdMarkerFound": {
    "$id": "#/definitions/eventNonAdMarkerFound",
    "required": [

```

```
    "eventType",
    "adMarkers"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "NON_AD_MARKER_FOUND"
    }
  }
},
"eventMakingAdsRequest": {
  "$id": "#/definitions/eventMakingAdsRequest",
  "required": [
    "eventType",
    "adsRequestUrl"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "MAKING_ADS_REQUEST"
    }
  }
},

"eventModifiedTargetUrl": {
  "$id": "#/definitions/eventModifiedTargetUrl",
  "required": [
    "eventType",
    "originalTargetUrl",
    "updatedTargetUrl"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "MODIFIED_TARGET_URL"
    }
  }
},

"eventRawAdsResponse": {
  "$id": "#/definitions/eventRawAdsResponse",
  "required": [
    "eventType",
    "rawAdsResponse",
```

```

    "rawAdsResponseIndex"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "RAW_ADS_RESPONSE"
    }
  }
},

```

```

"eventVastResponse": {
  "$id": "#/definitions/eventVastResponse",
  "_comment": "NOTE: the vastResponse property should ideally be marked as a
required field for this event, but unfortunately, in the case of an empty vast
response, we currently emit an EMPTY_VAST_RESPONSE followed by a VAST_RESPONSE, and
the vastResponse property is not present in the latter. We need to fix this so that we
don't emit both of those events in the empty response case, and update this schema to
flag vastResponse as required for VAST_RESPONSE.",

```

```

  "required": [
    "eventType"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "VAST_RESPONSE"
    }
  }
},

```

```

"eventFilledAvail": {
  "$id": "#/definitions/eventFilledAvail",
  "required": [
    "eventType",
    "avail"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "FILLED_AVAIL"
    }
  }
},

```

```

"eventFilledOverlayAvail": {

```

```
    "$id": "#/definitions/eventFilledOverlayAvail",
    "required": [
      "eventType",
      "avail"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "FILLED_OVERLAY_AVAIL"
      }
    }
  },

  "eventErrorVastMissingOverlays": {
    "$id": "#/definitions/eventErrorVastMissingOverlays",
    "required": [
      "eventType",
      "adsRequestUrl",
      "requestHeaders"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_VAST_MISSING_OVERLAYS"
      }
    }
  },

  "eventErrorFiringBeaconFailed": {
    "$id": "#/definitions/eventErrorFiringBeaconFailed",
    "required": [
      "eventType",
      "error",
      "beaconInfo"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_FIRING_BEACON_FAILED"
      }
    }
  },

  "eventWarningNoAdvertisements": {
```



```
    "$id": "#/definitions/eventWarningNoAdvertisements",
    "required": [
      "eventType"
    ],
    "_comment": "We should really have a more descriptive error field for these
events",
    "properties": {
      "eventType": {
        "type": "string",
        "const": "WARNING_NO_ADVERTISEMENTS"
      }
    }
  },

  "eventUnknownHost": {
    "$id": "#/definitions/eventUnknownHost",
    "required": [
      "eventType",
      "requestHeaders"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_UNKNOWN_HOST"
      }
    }
  },

  "eventErrorAdsTimeout": {
    "$id": "#/definitions/eventErrorAdsTimeout",
    "required": [
      "eventType",
      "adsRequestUrl",
      "requestHeaders"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_ADS_TIMEOUT"
      }
    }
  },

  "eventPlannedAvail": {
```

```
    "$id": "#/definitions/eventPlannedAvail",
    "required": [
      "eventType"
    ],
    "_comment": "TODO: Flesh this out as we implement it",
    "properties": {
      "eventType": {
        "type": "string",
        "const": "PLANNED_AVAIL"
      }
    }
  },

  "eventEmptyVastResponse": {
    "$id": "#/definitions/eventEmptyVastResponse",
    "required": [
      "eventType"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "EMPTY_VAST_RESPONSE"
      }
    }
  },

  "eventEmptyVmapResponse": {
    "$id": "#/definitions/eventEmptyVmapResponse",
    "required": [
      "eventType"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "EMPTY_VMAP_RESPONSE"
      }
    }
  },

  "eventErrorUnknown": {
    "$id": "#/definitions/eventErrorUnknown",
    "required": [
      "eventType"
    ],
```

```

    "_comment": "TODO: we should have a field for the exception message or
something",
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_UNKNOWN"
      }
    }
  },

  "eventVastRedirect": {
    "$id": "#/definitions/eventVastRedirect",
    "required": [
      "eventType"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "VAST_REDIRECT"
      }
    }
  },

  "eventRedirectedVastResponse": {
    "$id": "#/definitions/eventRedirectedVastResponse",
    "required": [
      "eventType"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "REDIRECTED_VAST_RESPONSE"
      }
    }
  },
  "_comment": "NOTE that the property vastResponse is not required because empty
vast responses do not contain a vastResponse."
},

  "eventErrorAdsResponseParse": {
    "$id": "#/definitions/eventErrorAdsResponseParse",
    "required": [
      "eventType"
    ],
    "_comment": "We should have a field with an error message here",

```

```
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_ADS_RESPONSE_PARSE"
      }
    }
  },

  "eventErrorAdsInvalidResponse": {
    "$id": "#/definitions/eventErrorAdsInvalidResponse",
    "required": [
      "eventType",
      "additionalInfo"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_ADS_INVALID_RESPONSE"
      }
    }
  },

  "eventErrorAdsMissingImpression": {
    "$id": "#/definitions/eventErrorAdsMissingImpression",
    "required": [
      "eventType"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_VAST_MISSING_IMPRESSION"
      }
    }
  },

  "eventErrorDisallowedHost": {
    "$id": "#/definitions/eventErrorDisallowedHost",
    "required": [
      "eventType"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_DISALLOWED_HOST"
      }
    }
  }
}
```

```
    }
  }
},

"eventPersonalizationDisabled": {
  "$id": "#/definitions/eventPersonalizationDisabled",
  "required": [
    "eventType"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "ERROR_PERSONALIZATION_DISABLED"
    }
  }
},

"eventWarningDynamicVariableSubFailed": {
  "$id": "#/definitions/eventWarningDynamicVariableSubFailed",
  "required": [
    "eventType",
    "adsRequestUrl"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "WARNING_URL_VARIABLE_SUBSTITUTION_FAILED"
    }
  }
},

"eventVodTimeBasedAvailPlanVastResponseForOffset": {
  "$id": "#/definitions/eventVodTimeBasedAvailPlanVastResponseForOffset",
  "required": [
    "eventType",
    "vastResponse"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "VOD_TIME_BASED_AVAIL_PLAN_VAST_RESPONSE_FOR_OFFSET"
    }
  }
},
```

```

"eventVodTimeBasedAvailPlanSuccess": {
  "$id": "#/definitions/eventVodTimeBasedAvailPlanSuccess",
  "required": [
    "eventType",
    "vodCreativeOffsets"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "VOD_TIME_BASED_AVAIL_PLAN_SUCCESS"
    }
  }
},

"creativeUniqueId": {
  "type": "string",
  "description": "The unique identifier for the ad, used as a key for transcoding.
This is the ID field for the creative in the VAST response, if available. Otherwise,
it's the mezzanine URL of the ad. "
},

"adSystem": {
  "type": "string",
  "description": "The value of the <code>AdSystem</code> tag in the VAST response.
"
},

"vastDuration": {
  "type": "number",
  "description": "The duration of the ad, as parsed from the VAST response."
},

"transcodedAdDuration": {
  "type": "number",
  "description": "The duration of the ad, calculated from the transcoded asset."
},

"adContent": {
  "$id": "#/properties/adContent",
  "type": ["object", "null"],
  "title": "adContent",
  "description": "Information about the content of the inserted ad.",
  "additionalProperties": false,

```

```

    "properties": {
      "adPlaylistUri": {
        "$id": "#/properties/adContent/adPlaylistUri",
        "type": "object",
        "title": "adPlaylistUri",
        "description": "The mapping from the origin manifest for a variant to the ad
manifest for the variant. For DASH, this contains a single entry, because all variants
are represented in a single DASH manifest. ",
        "additionalProperties": {
          "$id": "#/properties/adContent/adPlaylistUri/adPlaylistUri",
          "type": "string",
          "description": "The URL of the ad manifest for the specific variant."
        }
      }
    }
  },

  "adMezzanineUri": {
    "type": "string",
    "description": "The URL of the mezzanine version of the ad, which is the input to
the transcoder."
  },

  "bitrate": {
    "type": "integer",
    "examples": [
      533
    ],
    "description": "The bitrate."
  },

  "width": {
    "type": "integer",
    "examples": [
      1280
    ],
    "description": "Width in pixels."
  },

  "height": {
    "type": "integer",
    "examples": [
      720
    ],
    "description": "Height in pixels."
  }
},

```

```

"trackingEvents": {
  "type": "object",
  "title": "trackingEvents",
  "description": "The tracking beacon URLs for the various tracking events for the
ad. The keys are the event names, and the values are a list of beacon URLs.",

  "additionalProperties": {
    "type": "array",
    "description": "The list of beacon URLs for the specified tracking event
(impression, complete, and so on)",
    "items": {
      "type": "string",
      "description": "The beacon URLs for this tracking event."
    }
  }
},

"nonLinearAds": {
  "$id": "#/properties/nonLinearAds",
  "type": "object",
  "title": "nonLinearAds",
  "description": "A NonLinearAds as it appears in the VAST response.",
  "required": [
    "nonLinearAdList",
    "nonLinearTrackingEvents"
  ],
  "properties": {
    "nonLinearAdList": {
      "type": "array",
      "description": "List of non linear ads as they exist within one
NonLinearAds.",
      "items": {
        "type": "object",
        "title": "nonLinearAdList",
        "description": "List of NonLinearAd as they are parsed from its parent
NonLinearAds.",
        "properties": {
          "nonLinearAdId": {
            "type": "string",
            "description": "Ad ID of this non linear ad."
          },
          "nonLinearAdSystem": {
            "type": "string",

```



```
    "description": "Ad system of this non linear ad's parent Inline ad."
  },
  "nonLinearAdTitle": {
    "type": "string",
    "description": "Ad title of this non linear ad's parent Inline ad."
  },
  "nonLinearCreativeId": {
    "type": "string",
    "description": "Creative ID of this non linear ad's parent Creative
ad."
  },
  "nonLinearCreativeAdId": {
    "type": "string",
    "description": "Creative ad ID of this non linear ad."
  },
  "nonLinearCreativeSequence": {
    "type": "string",
    "description": "Creative sequence of this non linear ad."
  },
  "nonLinearWidth": {
    "type": "string",
    "description": "Width of this non linear ad."
  },
  "nonLinearHeight": {
    "type": "string",
    "description": "Height of this non linear ad."
  },
  "nonLinearExpandedWidth": {
    "type": "string",
    "description": "Expanded width of this non linear ad."
  },
  "nonLinearExpandedHeight": {
    "type": "string",
    "description": "Expanded height of this non linear ad."
  },
  "nonLinearScalable": {
    "type": "boolean",
    "description": "Boolean denoting if this non linear ad is scalable."
  },
  "nonLinearMaintainAspectRatio": {
    "type": "boolean",
    "description": "Boolean denoting if aspect ratio should be maintained
for this non linear ad."
  },
  },
```

```

    "nonLinearMinSuggestedDuration": {
      "type": "string",
      "description": "Min suggested duration for this non linear ad."
    },
    "nonLinearApiFramework": {
      "type": "string",
      "description": "API framework for this non linear ad's parent Inline
ad."
    },
    "nonLinearStaticResource": {
      "type": "string",
      "description": "Static resource for this non linear ad."
    },
    "nonLinearStaticResourceCreativeType": {
      "type": "string",
      "description": "Static Resource creative type for this non linear ad."
    },
    "nonLinearIFrameResource": {
      "type": "string",
      "description": "I-Frame resource for this non linear ad."
    },
    "nonLinearHtmlResource": {
      "type": "string",
      "description": "HTML resource for this non linear ad."
    },
    "nonLinearAdParameters": {
      "type": "string",
      "description": "Ad parameters for this non linear ad."
    },
    "nonLinearClickThrough": {
      "type": "string",
      "description": "Click Through data for this non linear ad."
    },
    "nonLinearClickTracking": {
      "type": "string",
      "description": "Click Tracking data for this non linear ad."
    },
    "nonLinearClickTrackingId": {
      "type": "string",
      "description": "Click Tracking ID for this non linear ad."
    }
  }
}
},

```

```

    "nonLinearTrackingEvents": { "$ref": "#/definitions/trackingEvents" },
    "extensions": {
      "$id": "#/properties/nonLinearAds/extensions",
      "type": "array",
      "description": "Extensions that exist for this NonLinearAds.",
      "items": {
        "$id": "#/properties/nonLinearAds/extensions/items",
        "type": "object",
        "title": "Extensions",
        "description": "Extensions found in non linear ads",
        "additionalProperties": false,
        "properties": {
          "extensionType": {
            "$id": "#/properties/nonLinearAds/extensions/extensionType",
            "type": "string",
            "description": "The value of the extension type attribute of the
<code>Extensions</code> tag.",
            "examples": [
              "FreeWheel"
            ]
          },
          "extensionContent": {
            "$id": "#/properties/nonLinearAds/extensions/extensionContent",
            "type": "string",
            "description": "The extension content attribute of the
<code>Extensions</code> tag.",
            "examples": [
              "progressive"
            ]
          }
        }
      }
    },
    "adBreakTrackingEvents": {
      "$id": "#/properties/adBreakTrackingEvents",
      "type": "object",
      "title": "adBreakTrackingEvents",
      "description": "These are all VMAP ad break tracking events.",
      "additionalProperties": {
        "type": "array",
        "description": "VMAP/ad break tracking events and corresponding URL",
        "items": {

```

```

        "type": "string",
        "description": "The beacon URLs for this tracking event."
    }
},
"vastAd": {
    "$id": "#/properties/vastAd",
    "type": "object",
    "title": "vastAd",
    "description": "Information about a single ad parsed from the VAST response.",
    "required": [
        "vastAdId",
        "adSystem",
        "adTitle",
        "creativeId",
        "creativeAdId",
        "duration",
        "vastMediaFiles",
        "trackingEvents"
    ],
    "additionalProperties": false,
    "properties": {
        "vastAdId": {
            "$id": "#/properties/vastAd/vastAdId",
            "type": "string",
            "description": "The value of the id attribute of the <code>Ad</code> tag in
the VAST response",
            "examples": [
                "ad1"
            ]
        },
        "adSystem": {"$ref": "#/definitions/adSystem" } ,
        "adTitle": {
            "$id": "#/properties/vastAd/adTitle",
            "type": "string",
            "description": "The media files that are available for the ad in the VAST
response.",
            "examples": [
                "External NCA1C1L1 LinearInlineSkippable"
            ]
        },
        "creativeId": {
            "$id": "#/properties/vastAd/creativeId",
            "type": "string",

```

```

    "description": "The value of the id attribute of the <code>Creative</code>
tag in the VAST response.",
    "examples": [
        "creative1"
    ]
},
"creativeAdId": {
    "$id": "#/properties/vastAd/creativeAdId",
    "type": "string",
    "description": "The value of the adId attribute of the <code>Creative</code>
tag in the VAST response."
},
"duration": {
    "$id": "#/properties/vastAd/duration",
    "type": "number",
    "description": "The approximate duration of the ad, based on the
<code>duration</code> tag in the <code>linear</code> element of the VAST response.",
    "examples": [
        30,
        30.0
    ]
},
"vastMediaFiles": {
    "$id": "#/properties/vastAd/vastMediaFiles",
    "type": "array",
    "description": "The list of available media files for the ad in the VAST
response.",
    "items": {
        "$id": "#/properties/vastAd/vastMediaFiles/items",
        "type": "object",
        "title": "vastMediaFile",
        "description": "Information about a media file for the ad.",
        "required": [
            "uri",
            "id",
            "delivery",
            "type",
            "apiFramework",
            "width",
            "height",
            "bitrate"
        ],
        "additionalProperties": false,
        "properties": {

```

```

    "uri": { "$ref": "#/definitions/adMezzanineUri" },
    "id": {
      "$id": "#/properties/vastAd/vastMediaFiles/items/properties/id",
      "type": "string",
      "description": "The value of the id attribute of the <code>MediaFile</code> tag.",
      "examples": [
        "GDFP"
      ]
    },
    "delivery": {
      "$id": "#/properties/vastAd/vastMediaFiles/items/properties/delivery",
      "type": "string",
      "description": "The protocol used for the media file, set to either progressive or streaming.",
      "examples": [
        "progressive"
      ]
    },
    "type": {
      "$id": "#/properties/vastAd/vastMediaFiles/items/properties/type",
      "type": "string",
      "description": "The MIME type of the media file, taken from the type attribute of the <code>MediaFile</code> tag.",
      "examples": [
        "video/mp4"
      ]
    },
    "apiFramework": {
      "$id": "#/properties/vastAd/vastMediaFiles/items/properties/apiFramework",
      "type": "string",
      "description": "The API framework needed to manage the media file. Example: <code>VPAID</code>."
    },
    "width": {
      "$ref": "#/definitions/width"
    },
    "height": {
      "$ref": "#/definitions/height"
    },
    "bitrate": {
      "$ref": "#/definitions/bitrate"
    }
  }

```

```
    }
  }
},
"trackingEvents": { "$ref": "#/definitions/trackingEvents" },
"vastAdTagUri": {
  "$id": "#/properties/vastAd/vastAdTagUri",
  "type": "string",
  "description": "The VMAP-specific redirect URI for an ad.",
  "examples": [
    "https://ads.redirect.com/redirect1"
  ]
}
}
}
}
```

AWS Elemental MediaTailor Beschreibung und Ereignistypen der Manifestprotokolle

In den folgenden Abschnitten werden die Protokolle beschrieben, die ausgegeben MediaTailor werden, um Ereignisse mit dem Ursprungsserver zu beschreiben, wenn ein Manifest angefordert und empfangen wird. Dies sind ManifestService Protokolle.

Themen

- [ManifestService Veranstaltungen](#)
- [Das Manifest protokolliert die Eigenschaften](#)

ManifestService Veranstaltungen

Die folgenden Ereignisse werden bei MediaTailor Interaktionen mit dem Ursprung ausgelöst.

Protokoll	Beschreibung
CONFIG_SECURITY_ERROR	Die MediaTailor Konfiguration hat ein Sicherheitsproblem.
CONFIG_SYNTAX_ERROR	Der Ursprung und der Asset-Pfad führen zu einer falsch formatierten URL.

Protokoll	Beschreibung
CONNECTION_ERROR	Die MediaTailor Verbindung zum Ursprung wurde abgelehnt oder ist fehlgeschlagen.
GENERATED_MANIFEST	MediaTailor hat ein Manifest generiert. Sie müssen den Debug-Modus aktiviert haben, um diese Protokolle empfangen zu können. Hinweise zum Debug-Protokollmodus, einschließlich seiner Aktivierung, finden Sie unter. Debug-Logs generieren
HOST_DISALLOWED	MediaTailor lässt keine HTTP-Anfragen an diesen Host zu.
INCOMPATIBLE_HLS_VERSION	Das Manifest verwendet eine inkompatible HLS-Version. MediaTailor erfordert Version 3 oder höher.
INVALID_SINGLE_PERIOD_DASH_MANIFEST	Das DASH-Manifest mit einer Periode ist ungültig. MediaTailor durchläuft ein DASH-Manifest mit einer einzigen Periode.
IO_ERROR	MediaTailor ist bei der Kommunikation mit dem Ursprung auf einen IO-Fehler gestoßen.
LAST_PERIOD_MISSING_AUDIO	Im letzten Abschnitt des DASH-Manifests fehlt der gesamte Ton <code>AdaptationSets</code> aufgrund einer falschen Ausrichtung der ursprünglichen Audio- oder Videodaten. Um Probleme bei der Wiedergabe zu vermeiden, sollten Sie die Veröffentlichung des letzten Zeitraums bis mindestens zur nächsten Anfrage verschieben.

Protokoll	Beschreibung
LAST_PERIOD_MISSING_AUDIO_WARNING	Im letzten Abschnitt des DASH-Manifests fehlt <code>AdaptationSets</code> aufgrund einer falschen Ausrichtung der ursprünglichen Audio- oder Videodaten das gesamte Audio. Es wird entschieden, den letzten Zeitraum zu veröffentlichen (nicht zu verzögern). Fehlendes Audio kann zu Problemen bei der Wiedergabe führen.
MANIFEST_ERROR	Die MediaTailor Manifestanforderung ist fehlgeschlagen.
NO_MASTER_OR_MEDIA_PLAYLIST	Die ursprüngliche Antwort enthält keine primäre Playlist oder Medien-Playlist.
NO_MASTER_PLAYLIST	Die ursprüngliche Antwort enthält nicht die erwartete primäre Playlist.
NO_MEDIA_PLAYLIST	Die ursprüngliche Antwort enthält nicht die erwartete Medien-Playlist.
ORIGIN_MANIFEST	MediaTailor hat ein Ursprungsmanifest abgerufen. Sie müssen den Debug-Modus aktiviert haben, um diese Protokolle empfangen zu können. Hinweise zum Debug-Protokollmodus, einschließlich seiner Aktivierung, finden Sie unter Debug-Logs generieren .
PARSING_ERROR	Der Ursprung ist nicht in der Lage, die Manifestanforderung zu analysieren.
SCTE35_PARSING_ERROR	MediaTailor kann das <code>Signal Binary Element</code> im Manifest nicht analysieren.

Protokoll	Beschreibung
SESSION_INITIALIZED	Eine Sitzung wurde initialisiert. Sie müssen den Debug-Modus aktiviert haben, um diese Protokolle empfangen zu können. Hinweise zum Debug-Protokollmodus, einschließlich seiner Aktivierung, finden Sie unter. Debug-Logs generieren
TIMEOUT_ERROR	Bei der MediaTailor Manifestanforderung wurde das Zeitlimit überschritten.
TRACKING_RESPONSE	MediaTailor hat eine Tracking-Antwort zugestellt. Sie müssen den Debug-Modus aktiviert haben, um diese Protokolle empfangen zu können. Hinweise zum Debug-Protokollmodus, einschließlich seiner Aktivierung, finden Sie unter. Debug-Logs generieren
UNKNOWN_ERROR	MediaTailor ist auf einen unbekannten Fehler gestoßen.
UNKNOWN_HOST	Der Host ist unbekannt.
UNSUPPORTED_SINGLE_PERIOD_DASH_MANIFEST	Das DASH-Manifest mit einer Periode wird nicht unterstützt. MediaTailor durchläuft ein DASH-Manifest mit einer einzigen Periode.

Das Manifest protokolliert die Eigenschaften

In diesem Abschnitt werden die Eigenschaften der Manifestprotokolle beschrieben.

Eigenschaft	Typ	Erforderlich
awsAccountId	Zeichenfolge	true

Eigenschaft	Typ	Erforderlich
eventTimestamp	Zeichenfolge	true
originId	Zeichenfolge	true
customerId	Zeichenfolge	false
eventType	Zeichenfolge	false
sessionId	Zeichenfolge	false
originRequestUrl	Zeichenfolge	false
mediaTailorPath	Zeichenfolge	false
requestId	Zeichenfolge	false
responseBody	Zeichenfolge	false
sessionType	Zeichenfolge (zulässige Werte: [DASH, HLS])	false
requestNextToken	Zeichenfolge	false
eventDescription	Zeichenfolge	false
assetPath	Zeichenfolge	false
originFullUrl	Zeichenfolge	false
originPrefixUrl	Zeichenfolge	false
additionalInfo	Zeichenfolge	false
cause	Zeichenfolge	false
response	Zeichenfolge	false
httpCode	Zeichenfolge	false
errorMessage	Zeichenfolge	false

Eigenschaft	Typ	Erforderlich
adAdsResponse	Zeichenfolge	false
adAdsRawResponse	Zeichenfolge	false
adAdsRequest	Zeichenfolge	false
adNumNewAvails	Zeichenfolge	false
generatedMediaPlay list	Zeichenfolge	false
requestStartTime	Zeichenfolge	false
requestEndTime	Zeichenfolge	false
requestStartTimeEp ochMillis	Zeichenfolge	false
requestEndTimeEpoc hMillis	Zeichenfolge	false

AWS Elemental MediaTailor Beschreibung und Ereignistypen der Transcodierungsprotokolle

In den folgenden Abschnitten werden die Logs beschrieben, die ausgegeben werden MediaTailor , um Ereignisse mit dem Transcode-Dienst zu beschreiben, wenn Creatives auf das Anzeigen-Stitching vorbereitet werden. Dies sind Protokolle. TranscodeService

Themen

- [TranscodeService Veranstaltungen](#)
- [Eigenschaften von Transcode-Protokollen](#)

TranscodeService Veranstaltungen

Die folgenden Ereignisse werden bei MediaTailor Interaktionen während der Transcodierung von Anzeigen ausgelöst.

Protokoll	Beschreibung
IMPORT_ERROR	MediaTailor ist während eines Importauftrags (für vorkonditionierte Werbung) auf einen internen Fehler gestoßen. Es wurde ein leerer Satz von Anzeigen verwendet.
INITIALIZED	MediaTailor hat entweder einen Transcodierungsauftrag oder einen Importauftrag (für vorkonditionierte Anzeigen) initialisiert.
INTERNAL_ERROR	MediaTailor ist auf einen internen Fehler gestoßen. Es wurde ein leerer Satz von Anzeigen verwendet.
MISSING_VARIANTS	MediaTailor konnte die Anzeige aufgrund fehlender Varianten nicht transkodieren. Es wurde ein leerer Satz von Anzeigen verwendet.
PROFILE_NOT_FOUND	MediaTailor konnte die Anzeige nicht transkodieren, da ein zu transkodierendes Profil fehlt. Es wurde ein leerer Satz von Anzeigen verwendet.
TRANSCODE_COMPLETED	Die Videotranskodierung ist abgeschlossen. Die Anzeige kann zum Einfügen von Anzeigen verwendet werden.
TRANSCODE_ERROR	MediaTailor ist während eines Transcodierungsauftrags auf einen internen Fehler gestoßen. Es wurde ein leerer Satz von Anzeigen verwendet.
TRANSCODE_IN_PROGRESS	Die Videotranskodierung ist im Gange. Das transkodierte Video ist nicht bereit. Es wird ein leerer Satz von Anzeigen verwendet.

Eigenschaften von Transcode-Protokollen

In diesem Abschnitt werden die Eigenschaften der Transcodierungsprotokolle beschrieben.

Eigenschaft	Typ	Erforderlich	Beschreibung
<code>awsAccountId</code>	Zeichenfolge	true	Die AWS Konto-ID für die MediaTailor Konfiguration, die für die Sitzung verwendet wurde.
<code>eventTimestamp</code>	Zeichenfolge	true	Das Datum und die Uhrzeit des Ereignisses.
<code>originId</code>	Zeichenfolge	true	Der Konfigurationsname aus der MediaTailor Konfiguration. Dies unterscheidet sich von der Quelle der Videoinhalte, die ebenfalls Teil der Konfiguration ist.
<code>eventType</code>	Zeichenfolge	false	Der Code für das Ereignis, das diese Protokollnachricht ausgelöst hat. Beispiel: <code>TRANSCODE_ERROR</code> .
<code>eventDescription</code>	Zeichenfolge	false	Eine kurze Beschreibung des Ereignisses, das diese Protokollnachricht ausgelöst hat. Sie wird vom MediaTailor Dienst

Eigenschaft	Typ	Erforderlich	Beschreibung
			bereitgestellt. Standardmäßig ist dieses Feld leer.
<code>sessionId</code>	Zeichenfolge	false	Die eindeutige numerische Kennung, die MediaTailor der Spiellersitzung zugewiesen wurde. Alle Anforderungen, die ein Spieler für eine Sitzung stellt, haben dieselbe Sitzungs-ID. Beispiel: e039fd39-09f0-46b2-aca9-9871cc116cde .
<code>creativeUniqueId</code>	Zeichenfolge	false	Die eindeutige Kennung für das Werbematerial, das transkodiert wird.
<code>profileName</code>	Zeichenfolge	false	
<code>adUri</code>	Zeichenfolge	false	Die URI für das Werbemittel.
<code>transcodeRequestId</code>	Zeichenfolge	false	Die eindeutige Kennung für diese Transcodierungsanforderung.

Eigenschaft	Typ	Erforderlich	Beschreibung
cacheStatus	Zeichenfolge	false	Gibt an, ob die MediaTailor transkodierte Anzeige zwischengespeichert wurde.

Verwenden von verkauften Protokollen zum Senden AWS Elemental MediaTailor von Protokollen

Sie können Vended-Logs verwenden, um flexibler zu MediaTailor sein und mehr Kontrolle darüber zu haben, wohin Logs aus Ihrer Wiedergabekonfiguration gesendet werden sollen.

MediaTailor Sendet bei verkauften Protokollen alle Protokollaktivitäten, die mit einer Konfiguration verknüpft sind, an Amazon CloudWatch Logs. CloudWatch Logs sendet dann den Prozentsatz der von Ihnen angegebenen Logs an das von Ihnen gewählte Ziel. Unterstützte Ziele sind eine Amazon CloudWatch Logs-Protokollgruppe, ein Amazon S3 S3-Bucket oder ein Amazon Data Firehose-Stream.

Da verkaufte Protokolle zu Mengenrabatten erhältlich sind, können Sie im Vergleich zum direkten Senden von Protokollen an Logs Kosteneinsparungen feststellen. CloudWatch Die Preise finden Sie unter Verkaufte Logs auf der Registerkarte Logs bei [Amazon CloudWatch Pricing](#).

Um Vended Logs zu verwenden, müssen Sie wie folgt vorgehen:

1. [Berechtigungen hinzufügen](#).
2. [Erstellen Sie Ziele für die Protokollzustellung](#).
3. [Konfigurieren Sie die Protokollzustellung in CloudWatch Logs](#).
4. [Aktivieren Sie Vended-Logs in MediaTailor](#).

Weitere Informationen zu Vended-Logs finden Sie unter [Aktivieren der Protokollierung von AWS Diensten](#) im CloudWatch Logs-Benutzerhandbuch. MediaTailor unterstützt Version 2 der verkauften Logs.

Schritt 1: Fügen Sie Berechtigungen für die MediaTailor Protokollzustellung hinzu

Die Person, die Versandprotokolle einrichtet, muss über die erforderlichen Berechtigungen verfügen, um das Zustellungsziel zu erstellen, die Protokollzustellung zu konfigurieren und die Anmeldung per Versand zu aktivieren. MediaTailor Verwenden Sie die folgenden Richtlinien, um sicherzustellen, dass Sie über die entsprechenden Berechtigungen zum Einrichten von Versandprotokollen verfügen.

Richtlinien für CloudWatch Protokolle und Lieferziele

Die folgenden Abschnitte im Amazon CloudWatch Logs-Benutzerhandbuch enthalten die Richtlinien, die es Ihnen ermöglichen, mit Protokollen in CloudWatch Logs und Ihren Lieferzielen zu arbeiten. Wenn Sie Protokolle an mehrere Standorte senden, können Sie die Richtlinienenerklärungen zu einer Richtlinie zusammenfassen, anstatt mehrere Richtlinien zu erstellen.

- [Protokolle, die an CloudWatch Logs gesendet wurden](#)
- [An Amazon S3 gesendete Protokolle](#)
- [An Firehose gesendete Logs](#)

Richtlinie für die Einrichtung über die Konsole

Wenn Sie den Versand von Verkaufsprotokollen über die Konsole statt über die API oder einrichten AWS CLI, müssen Sie in Ihrer Richtlinie über die folgenden zusätzlichen Berechtigungen verfügen.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowLogDeliveryActionsConsoleCWL",
      "Effect": "Allow",
      "Action": [
        "logs:DescribeLogGroups"
      ],
      "Resource": [
        "arn:aws:logs:us-east-1:111122223333:log-group:*"
      ]
    },
    {
      "Sid": "AllowLogDeliveryActionsConsoleS3",
      "Effect": "Allow",
      "Action": [
```

```

        "s3:ListAllMyBuckets",
        "s3:ListBucket",
        "s3:GetBucketLocation"
    ],
    "Resource": [
        "arn:aws:s3:::*"
    ]
},
{
    "Sid": "AllowLogDeliveryActionsConsoleFH",
    "Effect": "Allow",
    "Action": [
        "firehose:ListDeliveryStreams",
        "firehose:DescribeDeliveryStream"
    ],
    "Resource": [
        "*"
    ]
}
]
}

```

Die Richtlinie für ausgelieferte Logs meldet sich an MediaTailor

Um die Zustellung von ausgelieferten Protokollen in zu erstellen, anzuzeigen oder zu ändern MediaTailor, müssen Sie in Ihrer Richtlinie über die folgenden Berechtigungen verfügen.

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ServiceLevelAccessForLogDelivery",
            "Effect": "Allow",
            "Action": [
                "mediatailor:AllowVendedLogDeliveryForResource"],
            "Resource":
                "arn:aws:mediatailor:region:111122223333:playbackConfiguration/*"
        }
    ]
}

```

Informationen zum Hinzufügen von Berechtigungen und zum Arbeiten mit Richtlinien finden Sie unter [Identity and Access Management für AWS Elemental MediaTailor](#).

Schritt 2: Lieferziele für MediaTailor Protokolle erstellen

Erstellen Sie die Ressourcen, an die Ihre Protokolle gesendet werden sollen. Notieren Sie den ARN der Ressource, um ihn bei der Konfiguration der Protokollzustellung in einem späteren Schritt zu verwenden.

CloudWatch Protokolliert das Ziel der Protokollgruppenzustellung

Verwenden Sie eine der folgenden Optionen, um Hilfe bei der Erstellung einer Protokollgruppe zu erhalten.

- Informationen zur Konsole finden Sie unter [Erstellen einer Protokollgruppe in CloudWatch Logs](#) im Amazon CloudWatch Logs-Benutzerhandbuch.
- Informationen zur API finden Sie [CreateLogGroup](#) in der Amazon CloudWatch Logs API-Referenz.
- Informationen zu SDKs und CLI finden Sie unter [Verwendung CreateLogGroup mit einem AWS SDK oder AWS CLI](#) im Amazon CloudWatch Logs-Benutzerhandbuch.

Lieferziel für Amazon S3 S3-Buckets

Verwenden Sie eine der folgenden Optionen, um Hilfe bei der Erstellung eines Buckets zu erhalten.

- Informationen zur Konsole und zur CLI finden Sie unter [Create a Bucket](#) im Amazon Simple Storage Service-Benutzerhandbuch. SDKs
- Informationen zur API finden Sie [CreateBucket](#) in der Amazon Simple Storage Service API-Referenz.

Firehose Stream-Lieferziel

Hilfe beim Erstellen eines Streams finden Sie unter [Create a Firehose Stream from console](#) im Amazon Data Firehose Developer Guide.

Schritt 3: Aktivieren Sie Vended-Logs für die Wiedergabekonfiguration MediaTailor

Erstellen oder aktualisieren Sie die Wiedergabekonfiguration, mit der Protokolle an das Lieferziel gesendet werden, das Sie im vorherigen Schritt erstellt haben. Notieren Sie sich den Namen der Konfiguration für die Konfiguration der Protokollzustellung in einem späteren Schritt.

- Um die Übertragung von Protokollen über die Konsole zu aktivieren, verwenden [Konfiguration erstellen](#) oder [Eine Konfiguration bearbeiten](#) bearbeiten Sie eine Konfiguration, um auf die Protokollierungseinstellungen zuzugreifen. Wählen Sie für Logging-Strategien die Option Vended Logs aus.
- Um Vended Logs über die API zu aktivieren, müssen Sie über eine bestehende Konfiguration verfügen. Wird verwendet `ConfigureLogsForPlaybackConfiguration`, um die Protokollierungsstrategie Vended Logs hinzuzufügen.

Wenn Sie die bisherige MediaTailor Protokollierungsstrategie verwenden, bei der Logs direkt an Logs gesendet werden, und zu ausgeliehenen CloudWatch Logs migrieren möchten, finden Sie weitere Informationen unter [Migration der Protokollierungsstrategie](#).

 Important

Wenn Sie die Protokollstrategie von Legacy CloudWatch auf Vended Logs ändern, MediaTailor wird diese Änderung vorgenommen, sobald Sie die Aktualisierungen speichern. Sie werden keine Protokolle mehr empfangen, bis Sie die Protokollierung von Vended vollständig konfiguriert haben.

Schritt 4: Konfigurieren Sie die Protokollzustellung in CloudWatch Logs

In CloudWatch Logs müssen Sie drei Elemente erstellen, die die einzelnen Teile der Protokollzustellung darstellen. Diese Elemente werden in der Amazon CloudWatch Logs API-Referenz ausführlich beschrieben. [CreateDelivery](#) Die allgemeinen Schritte zur Konfiguration der Lieferung mit der CloudWatch Logs-API lauten wie folgt.

So konfigurieren Sie die Protokollzustellung in CloudWatch Logs (API)

1. Wird verwendet [PutDeliverySource](#), um die Quelle der Protokolle hinzuzufügen.

A `DeliverySource` steht für die Wiedergabekonfiguration, die die Protokolle generiert. Sie benötigen den Namen der Wiedergabekonfiguration, um die zu erstellen `DeliverySource`.

2. Wird verwendet [PutDeliveryDestination](#), um das Ziel hinzuzufügen, in das die Protokolle geschrieben werden.

A `DeliveryDestination` steht für das Lieferziel. Sie benötigen den ARN der Protokollgruppe, des Buckets oder des Streams, um den zu erstellen `DeliveryDestination`.

3. Verwenden Sie [PutDeliveryDestinationPolicy](#) diese Option, wenn Sie Logs kontenübergreifend bereitstellen.

Wenn sich das Lieferziel in einem anderen Konto als in der Wiedergabekonfiguration befindet, benötigen Sie ein `DeliveryDestinationPolicy`. Diese Richtlinie ermöglicht CloudWatch Logs die Übermittlung von Protokollen an die `DeliveryDestination`.

4. Wird verwendet [CreateDelivery](#), um das `DeliverySource` mit dem zu verknüpfen `DeliveryDestination`.

A `Delivery` steht für die Verbindung zwischen `DeliverySource` und `DeliveryDestination`.

Migrieren Sie Ihre AWS Elemental MediaTailor Logging-Strategie

Wenn Sie die Protokollstrategie von Legacy CloudWatch auf Vended Logs ändern, MediaTailor wird diese Änderung vorgenommen, sobald Sie die Updates speichern. Gehen Sie wie folgt vor, um Unterbrechungen in Ihrem Logging-Workflow zu vermeiden, um Ihre Protokollierungsstrategie zu migrieren.

1. Führen Sie die unter [Verwendung von verkauften Protokollen](#) beschriebenen Schritt aus. Wählen Sie für [Aktivieren Sie Vended-Logs in MediaTailor](#) beide Protokollierungsstrategien (Vending Logs und Legacy CloudWatch) aus.

MediaTailor sendet Protokolle sowohl über verkaufte Logs als auch direkt an CloudWatch Logs.

2. Nehmen Sie die erforderlichen Änderungen in Ihrem Arbeitsablauf vor, die von Ihrer Protokollierungsstrategie und Ihrem Lieferziel abhängen.
3. Besuchen Sie Legacy erneut [Aktivieren Sie Vended-Logs in MediaTailor](#) und entfernen Sie es CloudWatch aus den Protokollierungsstrategien.

AWS Elemental MediaTailor Logs direkt in Amazon CloudWatch Logs schreiben

MediaTailor erstellt Protokolle, die detaillierte Informationen über Sitzungsaktivitäten und Interaktionen mit dem Anzeigenentscheidungsserver enthalten, und schreibt sie an Amazon CloudWatch. Die Protokolle enthalten eine sequentielle Beschreibung der Aktivitäten, die während der Sitzung stattfinden.

MediaTailor kann auch verkaufte Holzscheite verwenden, um Flexibilität bei der Holzlieferung und Mengenrabatte zu erzielen. Informationen zu verkauften Protokollen finden Sie unter. [Verwendung von verkauften Protokollen](#)

Themen

- [Berechtigungen für Amazon CloudWatch Logs](#)
- [Protokoll „As Run“ für AWS Elemental MediaTailor Channel Assembly](#)
- [AWS Elemental MediaTailor ADS-Protokollanalyse in Amazon CloudWatch Logs Insights](#)

Berechtigungen für Amazon CloudWatch Logs

Verwenden Sie AWS Identity and Access Management (IAM), um eine Rolle zu erstellen, die AWS Elemental MediaTailor Zugriff auf Amazon CloudWatch gewährt. Sie müssen diese Schritte ausführen, damit CloudWatch Logs für Ihr Konto veröffentlicht werden. CloudWatch veröffentlicht automatisch Metriken für Ihr Konto.

Um den MediaTailor Zugriff zu ermöglichen CloudWatch

1. Öffnen Sie unter <https://console.aws.amazon.com/iam/> die IAM-Konsole.
2. Klicken Sie im Navigationsbereich der IAM-Konsole auf Rollen, und wählen Sie dann Rolle erstellen.
3. Wählen Sie den Rollentyp „Anderes AWS Konto“.
4. Geben Sie als Konto-ID Ihre AWS Konto-ID ein.
5. Wählen Sie Require external ID (Externe ID fordern) und geben Sie **Midas** ein. Mit dieser Option wird automatisch eine Bedingung zur Vertrauensrichtlinie hinzugefügt, die es dem Service nur dann ermöglicht, die Rolle anzunehmen, wenn in der Anforderung die richtige `sts:ExternalId` enthalten ist.
6. Wählen Sie Weiter: Berechtigungen aus.
7. Fügen Sie eine Berechtigungsrichtlinie hinzu, in der die Aktionen angegeben werden, die von dieser Rolle ausgeführt werden dürfen. Wählen Sie eine der folgenden Optionen aus und wählen Sie dann Next: Review (Nächster Schritt: Prüfen):
 - CloudWatchLogsFullAccessum vollen Zugriff auf Amazon CloudWatch Logs zu gewähren
 - CloudWatchFullAccessum vollen Zugriff auf Amazon zu gewähren CloudWatch
8. Geben Sie für Role name (Rollenname) den Namen **MediaTailorLogger** ein und klicken Sie auf Create role (Rolle erstellen).

9. Wählen Sie auf der Seite Roles (Rollen) die von Ihnen soeben erstellte Rolle aus.
10. Bearbeiten Sie die Vertrauensstellung, um den Prinzipal zu aktualisieren:
 1. Wählen Sie auf der Seite Summary (Übersicht) der Rolle die Registerkarte Trust relationship (Vertrauensstellung) aus.
 2. Wählen Sie Vertrauensstellung bearbeiten aus.
 3. Ändern Sie im Richtliniendokument den Principal in den MediaTailor Service. Das sollte wie folgt aussehen:

```
"Principal": {
  "Service": "mediatailor.amazonaws.com"
},
```

Die gesamte Richtlinie sollte folgendermaßen lauten:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "mediatailor.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:ExternalId": "Midas"
        }
      }
    }
  ]
}
```

4. Wählen Sie Update Trust Policy (Trust Policy aktualisieren).

Protokoll „As Run“ für AWS Elemental MediaTailor Channel Assembly

Das As Run-Protokoll in der CloudWatch MediaTailor/Channel/AsRunLog Protokollgruppe enthält Informationen zu Programmen und Werbeunterbrechungen, während sie abgespielt werden.

Wenn Sie einen Channel erstellen, ist das As Run-Protokoll standardmäßig deaktiviert. Mit der Konsole oder der AWS Command Line Interface (AWS CLI) können Sie den Protokollstatus „As Run“ für jeden Kanal in Ihrem Konto aktivieren und deaktivieren.

Wenn Sie das As-Run-Protokoll aktivieren, MediaTailor wird automatisch eine dienstbezogene Rolle erstellt, mit der Sie das As-Run-Protokoll in Ihrem CloudWatch Logs-Konto schreiben und verwalten können MediaTailor . Weitere Informationen zu serviceverknüpften Rollen finden Sie unter [Verwenden von serviceverknüpften Rollen für MediaTailor](#).

 Note

Das As Run Log unterstützt derzeit nur das Standardprogramm. Derzeit werden die durch Programmregeln erstellten AlternateMedia nicht unterstützt. Dies bedeutet, dass das As Run Log für AlternateMedia derzeit nicht generiert wird.

Themen

- [Das As Run-Protokoll aktivieren](#)
- [Das As Run-Protokoll deaktivieren](#)

Das As Run-Protokoll aktivieren

Um das As Run-Protokoll zu aktivieren, geben Sie den Kanalnamen an und aktivieren Sie den Protokolltyp As Run für diesen Kanal.

Console

Um das As Run-Protokoll beim Erstellen eines Kanals zu aktivieren

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie in der Navigationsleiste die Option Kanal erstellen aus.
4. Konfigurieren Sie Ihren Kanal in den Bereichen Kanaldetails festlegen, Ausgänge konfigurieren und Zugriffskontrolle wie gewünscht.
5. Wählen Sie im Bereich „Zugriffskontrolle“ die Option Weiter aus.

6. Wählen Sie im Bereich Protokollierung unter Protokolltypen die Option Als Ausführung aktivieren aus, um das Protokoll Als Ausführung zu aktivieren.

Um das As-Run-Protokoll bei der Aktualisierung eines Kanals zu aktivieren

 Note

Wenn der Kanal gerade läuft, müssen Sie ihn zuerst beenden, bevor Sie ihn aktualisieren können. Nachdem Sie den Kanal gestoppt haben, können Sie „Aktionen“ > „Bearbeiten“ wählen, um mit der Aktualisierung des Kanals zu beginnen.

1. Melde dich bei an AWS Management Console und öffne die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie den Kanal aus, für den Sie das As Run-Protokoll aktualisieren möchten.
4. Wählen Sie Actions (Aktionen) und Edit (Bearbeiten).
5. Aktualisieren Sie in den Bereichen Kanaldetails festlegen, Ausgänge konfigurieren und Zugriffskontrolle Ihre Kanalkonfiguration wie gewünscht.
6. Wählen Sie im Bereich „Zugriffskontrolle“ die Option Weiter aus.
7. Wählen Sie im Bereich Protokollierung unter Protokolltypen die Option Als Ausführung aktivieren aus, um das Protokoll Als Ausführung zu aktivieren.

Um das As-Run-Protokoll auf der Registerkarte Protokollierung zu aktivieren

 Note

Wenn der Channel gerade läuft, müssen Sie die Registerkarte Logging verwenden, anstatt Aktionen > Bearbeiten zu wählen, um das As Run-Protokoll zu aktivieren.

1. Melden Sie sich bei an AWS Management Console und öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie den Kanal aus, für den Sie das As Run-Protokoll aktivieren möchten.

4. Wählen Sie in der Navigationsleiste unter dem Namen des Kanals die Option Logging aus.
5. Wählen Sie unter Protokollierung > Protokolltypen die Option Als ausgeführt aus, um das Protokoll Als ausgeführt zu aktivieren.

AWS Command Line Interface (AWS CLI)

Um das As Run-Protokoll zu aktivieren

Führen Sie den [configure-logs-for-channel](#) Befehl aus und geben Sie die entsprechenden Werte für die erforderlichen Parameter an.

Dieses Beispiel ist für Linux, macOS oder Unix formatiert und verwendet den umgekehrten Schrägstrich (\) zur Verbesserung der Lesbarkeit.

```
$ aws mediatailor configure-logs-for-channel \  
--channel-name MyChannel \  
--log-types AS_RUN
```

Dieses Beispiel ist für Microsoft Windows formatiert und verwendet das Zeilenfortsetzungszeichen Caret (^), um die Lesbarkeit zu verbessern.

```
C:\> aws mediatailor configure-logs-for-channel ^  
--channel-name MyChannel ^  
--log-types AS_RUN
```

Wobei gilt:

- *MyChannel* ist der Name des Kanals, dessen Eigentümer Sie sind und für den Sie das As Run-Protokoll aktivieren möchten.

Wird der Befehl erfolgreich ausgeführt, erhalten Sie eine Ausgabe ähnlich der folgenden:

```
{  
  "ChannelName": "MyChannel",  
  "LogTypes": [  
    "AS_RUN"  
  ]  
}
```

Das As Run-Protokoll deaktivieren

Um das As Run-Protokoll für einen Kanal zu deaktivieren, für den es aktiviert ist, geben Sie den Kanalnamen an und deaktivieren Sie den Protokolltyp As Run für diesen Kanal.

Console

Um das As Run-Protokoll bei der Aktualisierung eines Kanals zu deaktivieren

Note

Wenn der Kanal gerade läuft, müssen Sie ihn zuerst beenden, bevor Sie ihn aktualisieren können. Nachdem Sie den Kanal gestoppt haben, können Sie „Aktionen“ > „Bearbeiten“ wählen, um mit der Aktualisierung des Kanals zu beginnen.

1. Melde dich bei an AWS Management Console und öffne die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie den Kanal aus, für den Sie das As Run-Protokoll aktualisieren möchten.
4. Wählen Sie Actions (Aktionen) und Edit (Bearbeiten).
5. Aktualisieren Sie in den Bereichen Kanaldetails festlegen, Ausgänge konfigurieren und Zugriffskontrolle Ihre Kanalkonfiguration wie gewünscht.
6. Wählen Sie im Bereich „Zugriffskontrolle“ die Option Weiter aus.
7. Deaktivieren Sie im Bereich Protokollierung unter Protokolltypen die Option Als Ausführung aktivieren, um das Protokoll Als Ausführung zu deaktivieren.

Um das As-Run-Protokoll auf der Registerkarte Protokollierung zu deaktivieren

Note

Wenn der Channel gerade läuft, müssen Sie die Registerkarte Logging verwenden, anstatt Aktionen > Bearbeiten zu wählen, um das As Run-Protokoll zu deaktivieren.

1. Melden Sie sich bei an AWS Management Console und öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.

2. Wählen Sie im Navigationsbereich Channel-Assembly > Channels aus.
3. Wählen Sie den Kanal aus, für den Sie das As Run-Protokoll deaktivieren möchten.
4. Wählen Sie in der Navigationsleiste unter dem Namen des Kanals die Option Logging aus.
5. Deaktivieren Sie unter Protokollierung > Protokolltypen die Option Als ausgeführt, um das Protokoll Als ausgeführt zu deaktivieren.

AWS Command Line Interface (AWS CLI)

Um das As-Run-Protokoll zu deaktivieren

Führen Sie den [configure-logs-for-channel](#) Befehl aus und geben Sie die entsprechenden Werte für die erforderlichen Parameter an.

Dieses Beispiel ist für Linux, macOS oder Unix formatiert und verwendet den umgekehrten Schrägstrich (\) zur Verbesserung der Lesbarkeit.

```
$ aws mediatailor configure-logs-for-channel \  
--channel-name MyChannel \  
--log-types
```

Dieses Beispiel ist für Microsoft Windows formatiert und verwendet das Zeilenfortsetzungszeichen Caret (^), um die Lesbarkeit zu verbessern.

```
C:\> aws mediatailor configure-logs-for-channel ^  
--channel-name MyChannel ^  
--log-types
```

Wobei gilt:

- *MyChannel* ist der Name des Kanals, dessen Eigentümer Sie sind und für den Sie das As Run-Protokoll deaktivieren möchten.

Wird der Befehl erfolgreich ausgeführt, erhalten Sie eine Ausgabe ähnlich der folgenden:

```
{  
  "ChannelName": "MyChannel",  
  "LogTypes": []
```

```
}
```

AWS Elemental MediaTailor ADS-Protokollanalyse in Amazon CloudWatch Logs Insights

Mit Amazon CloudWatch Logs Insights können Sie AWS Elemental MediaTailor ADS-Protokolle (Ad Decision Server) anzeigen und abfragen. MediaTailor sendet Ereignisprotokolle CloudWatch für normale Verarbeitungs- und Fehlerbedingungen an. Die Protokolle entsprechen einem JSON-Schema. Mit CloudWatch Logs Insights können Sie Protokolle nach Zeitrahmen auswählen und dann Abfragen für sie ausführen.

Allgemeine Informationen finden Sie unter [Analysieren von Protokolldaten mit CloudWatch Logs Insights](#).

Note

Um auf die Protokolle zugreifen zu können, benötigen Sie Berechtigungen für den Zugriff auf Amazon CloudWatch. Detaillierte Anweisungen finden Sie unter [Berechtigungen für Amazon CloudWatch Logs](#).

Um ADS-Protokolle über die CloudWatch Konsole anzuzeigen und abzufragen

1. Öffnen Sie die CloudWatch Konsole unter <https://console.aws.amazon.com/cloudwatch/>.
2. Wählen Sie im Navigationsbereich unter Logs, (Protokolle) die Option Insights(Einblicke) aus.
3. Geben Sie in der Suchleiste ein **AdDec**, und wählen Sie dann in der Dropdownliste aus **MediaTailor/AdDecisionServerInteractions**.
4. (Optional) Passen Sie den Zeitraum an, den Sie untersuchen möchten.
5. (Optional) Ändern Sie die Abfrage im Dialogfeld. Allgemeine Hinweise finden Sie unter [CloudWatch Logs Insights-Abfragesyntax](#). Beispiele für Abfragen für MediaTailor ADS finden Sie unter [Abfragen der ADS-Protokolle](#).
6. Wählen Sie Abfrage ausführen. Die Abfrage kann einige Sekunden dauern. Während dieser Zeit erscheint Cancel (Abbrechen) anstelle von Run query (Abfrage ausführen).
7. (Optional) Um die Ergebnisse als CSV-Datei zu exportieren, wählen Sie Actions (Aktionen) und dann Download query results (CSV) (Abfrageergebnisse herunterladen) aus.

 Note

Die Konsole begrenzt die Anzahl der Datensätze, die sie in Abfrageergebnissen zurückgibt und die sie exportiert. Verwenden Sie daher für Massendaten die API, das AWS Command Line Interface (AWS CLI) oder ein SDK.

Themen

- [Abfragen der ADS-Protokolle](#)

Abfragen der ADS-Protokolle

CloudWatch Logs Insights bietet eine Vielzahl von Optionen zum Abfragen Ihrer Logs. Ausführliche Informationen zur Abfragesyntax finden Sie unter [CloudWatch Logs Insights-Abfragesyntax](#). In diesem Abschnitt finden Sie Beispiele für gängige Abfragen für den Einstieg in Ihre ADS-Protokollabfragen. Alle Abfragen werden über die Protokolle für die aktuelle Zeitbereichseinstellung ausgeführt.

Die folgende Abfrage ruft alle Informationen aus den ADS-Protokollen ab.

```
fields @timestamp, eventType, sessionId, requestId, @message
| sort sessionId, @timestamp asc
```

Die folgende Abfrage ruft alle Anforderungen an den ADS ab. Diese Abfrage zeigt eine Möglichkeit, den Inhalt des Anforderungsheaders für MediaTailor Logs abzurufen.

```
fields @timestamp, adsRequestUrl, requestHeaders.0.value as @userAgent,
requestHeaders.1.value as @xForwardedFor, sessionId, requestId
| filter eventType = "MAKING_ADS_REQUEST"
| sort @timestamp asc
```

Die folgende Abfrage ruft die Anzeigen ab, die für eine bestimmte Sitzung MediaTailor eingefügt wurden.

```
fields @timestamp, sessionId, requestId, @message
| filter eventType = "FILLED_AVAIL"
| sort @timestamp asc
```

Die folgende Abfrage ruft das Tracking ab URLs , das im MediaTailor Namen des Spielers aufgerufen wurde.

```
fields @timestamp, beaconInfo.trackingEvent, beaconInfo.beaconUri,  
       beaconInfo.headers.0.value as @userAgent, beaconInfo.headers.1.value as  
       @xForwardedFor, sessionId, requestId  
| filter eventType = "BEACON_FIRED"  
| sort @timestamp asc
```

Die folgende Abfrage ruft Informationen für eine bestimmte Wiedergabebesitzung ab, indem die Ergebnisse nach sessionId gefiltert werden.

```
fields @timestamp, eventType, sessionId, requestId, @message  
| filter sessionId = "0aaf6507-c6f9-4884-bfe7-f2f841cb8195"  
| sort @timestamp asc
```

Die folgende Abfrage ruft Informationen für eine einzelne Anforderung ab, indem die Ergebnisse nach requestId gefiltert werden.

```
fields @timestamp, eventType, sessionId, requestId, @message  
| filter requestId = "f5d3cf39-6258-4cf1-b3f6-a34ff8bf641d"  
| sort @timestamp asc
```

Die folgende Abfrage ruft die Anzahl der Protokolleinträge für jeden Ereignistyp ab, der protokolliert wurde.

```
fields eventType  
| stats count() as @eventCount by eventType
```

Die folgende Abfrage ruft die Avail-ID und die Liste übersprungener Anzeigen für alle Avails ab, für die Anzeigen übersprungen wurden.

```
fields avail.availId  
| parse @message '"skippedAds":[*]' as @skippedAdsList  
| filter ispresent(@skippedAdsList)
```

Steuerung des Volumens der Protokolle AWS Elemental MediaTailor

MediaTailor Sitzungsprotokolle zum Einfügen von Anzeigen sind manchmal ausführlich. Um die Protokollkosten zu senken, können Sie den Prozentsatz der Sitzungsprotokolle definieren, die

MediaTailor an Amazon CloudWatch Logs gesendet werden. Wenn Ihre Wiedergabekonfiguration beispielsweise 1000 Sitzungen zum Einfügen von Werbeanzeigen umfasst und Sie einen aktivierten Prozentwert von festlegen60, werden Protokolle für 600 der Sitzungen an CloudWatch Logs MediaTailor gesendet. MediaTailor entscheidet nach dem Zufallsprinzip, für welche der Sitzungen Protokolle gesendet werden sollen. Wenn Sie Protokolle für eine bestimmte Sitzung anzeigen möchten, können Sie den [Debug-Protokollmodus](#) verwenden.

Wenn Sie einen Prozentsatz für die Protokollierung festlegen, MediaTailor wird automatisch eine mit dem Dienst verknüpfte Rolle erstellt, MediaTailor die die zum Schreiben von CloudWatch Protokollen in Ihr Konto erforderlichen Berechtigungen gewährt. Informationen zur MediaTailor Verwendung von dienstbezogenen Rollen finden Sie unter: [Verwenden von serviceverknüpften Rollen für MediaTailor](#)

Eine Protokollkonfiguration erstellen

Um den Prozentsatz der Sitzungsprotokolle zu kontrollieren, die in CloudWatch Protokolle MediaTailor geschrieben werden, erstellen Sie eine Protokollkonfiguration für Ihre Wiedergabekonfiguration. Wenn Sie eine Protokollkonfiguration erstellen, geben Sie einen Namen für die Wiedergabekonfiguration und einen aktivierten Prozentwert an.

Console

Um eine Protokollkonfiguration für eine bestehende Wiedergabekonfiguration zu erstellen

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Bereich Wiedergabekonfiguration die Wiedergabekonfiguration aus, für die Sie die Protokollkonfiguration festlegen möchten.
3. Wählen Sie Bearbeiten aus.
4. Geben Sie unter Protokollkonfiguration einen Wert für die Aktivierung in Prozent an.

Um eine Protokollkonfiguration für eine neue Wiedergabekonfiguration zu erstellen

- Folgen Sie dem Verfahren unter [Konfiguration protokollieren](#).

AWS Command Line Interface (AWS CLI)

Um eine Protokollkonfiguration für eine bestehende Wiedergabekonfiguration zu erstellen

Um eine Protokollkonfiguration mit dem zu erstellen AWS CLI, führen Sie den Befehl [configure-logs-for-playback-configuration](#) aus und geben Sie die entsprechenden Werte für die erforderlichen Parameter an.

Dieses Beispiel ist für Linux, macOS oder Unix formatiert und verwendet den umgekehrten Schrägstrich (\) zur Verbesserung der Lesbarkeit.

```
$ aws mediatailor configure-logs-for-playback-configuration \  
--percent-enabled 10 \  
--playback-configuration-name MyPlaybackConfiguration
```

Dieses Beispiel ist für Microsoft Windows formatiert und verwendet das Zeilenfortsetzungszeichen Caret (^), um die Lesbarkeit zu verbessern.

```
C:\> aws mediatailor configure-logs-for-playback-configuration ^  
--percent-enabled 10 ^  
--playback-configuration-name MyPlaybackConfiguration
```

Wobei gilt:

- *percent-enabled* ist der Prozentsatz der Sitzungsprotokolle zur Wiedergabekonfiguration, der an Logs gesendet wird. MediaTailor CloudWatch
- *playback-configuration-name* ist der Name der Wiedergabekonfiguration, für die die Protokollkonfigurationseinstellungen festgelegt werden sollen.

Wird der Befehl erfolgreich ausgeführt, erhalten Sie eine Ausgabe ähnlich der folgenden:

```
{  
  "PercentEnabled": 10,  
  "PlaybackConfigurationName": "MyPlaybackConfiguration"  
}
```

Um eine Protokollkonfiguration für eine neue Wiedergabekonfiguration zu erstellen

- Verwenden Sie die `configure-logs-for-playback-configuration` Option für den [put-playback-configuration](#) Befehl.

Deaktivierung einer Protokollkonfiguration

Nachdem Sie eine Protokollkonfiguration erstellt haben, können Sie sie nicht löschen — Sie können sie nur deaktivieren. Um die Protokollkonfiguration zu deaktivieren, setzen Sie in der MediaTailor Konsole oder API den Wert für die Aktivierung in Prozent auf 0. Dadurch wird die gesamte Sitzungsprotokollierung für diese Wiedergabekonfiguration deaktiviert.

Wenn Sie die dienstbezogene Rolle löschen möchten, die für die Protokollkonfiguration (en) in Ihrem Konto MediaTailor verwendet wird, müssen Sie zunächst alle Protokollkonfigurationen deaktivieren. Informationen zum Löschen der dienstverknüpften Rolle finden Sie unter. [Verwenden von serviceverknüpften Rollen für MediaTailor](#)

Console

So deaktivieren Sie die Protokollkonfiguration für eine Wiedergabekonfiguration

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie im Bereich Wiedergabekonfiguration die Wiedergabekonfiguration aus, für die Sie die Protokollkonfiguration deaktivieren möchten.
3. Wählen Sie Bearbeiten aus.
4. Stellen Sie unter Protokollkonfiguration den Wert für Prozent aktiviert auf ein 0. Dadurch wird die gesamte Sitzungsprotokollierung für diese Wiedergabekonfiguration deaktiviert.
5. Wählen Sie Speichern.

AWS Command Line Interface (AWS CLI)

Um eine Protokollkonfiguration zu deaktivieren

- Stellen Sie den percent-enabled Wert 0 mit dem Befehl [configure-logs-for-playback-configuration](#) auf ein.

Filterung von AWS Elemental MediaTailor Protokollen und Ereignissen

Protokolle, die von einer Wiedergabekonfiguration in ausgegeben werden, MediaTailor enthalten Informationen über eine Vielzahl von Aktivitäten, die während der Wiedergabesitzung stattfinden. Diese Aktivitäten werden im Ereignistyp der Protokolle identifiziert. Viele Ereignisse werden

standardmäßig protokolliert. Um die Kosten für Logs in Amazon unter Kontrolle zu halten CloudWatch, können Sie angeben, welche Logs ausgegeben MediaTailor werden.

MediaTailor bietet Ihnen die Kontrolle über die Protokollfilterung, sodass Sie Folgendes tun können:

- Geben Sie die Protokollereignisse an, die Sie aus den Protokollen ausschließen möchten
- Aktivieren Sie die Protokollierung von Rohantworten vom AD Decision Server (ADS)

Sie können diese Einstellungen für die Protokollfilterung unabhängig für jede Wiedergabesitzung oder als Standard für alle Wiedergabesitzungen für eine Wiedergabekonfiguration festlegen.

- Um Protokolle pro Sitzung zu filtern, fügen Sie Abfrageparameter in die Initialisierungsanfrage für die Wiedergabe-Sitzung ein.
- Um Protokolle auf Basis der Konfiguration pro Wiedergabe zu filtern, verwenden Sie die MediaTailor Konsole oder die API, um Ihre Präferenzen in den Einstellungen für die Wiedergabekonfiguration anzugeben.

Die folgenden Abschnitte enthalten Anweisungen zur Aktivierung der Protokollfilterung für Sitzungen und Wiedergabekonfigurationen.

Protokollfilter pro Sitzung

Um für jede Sitzung eine benutzerdefinierte Ebene der Protokolldetails zu definieren, fügen Sie die folgenden Parameter an Ihre erste server- oder clientseitige Wiedergabe-Sitzungsanforderung an. Fügen Sie den Parametern Werte hinzu, um die Ereignisse darzustellen, die Sie ein- oder ausschließen möchten, und zwar in einem kommasetrennten Format:

- `aws.adsInteractionLogPublishOptInEventTypesum` Protokolle für bestimmte Interaktionen mit dem AD Decision Server (ADS) zu erhalten.
- `aws.adsInteractionLogExcludeEventTypesum` den Empfang von Protokollen für bestimmte ADS-Interaktionen zu beenden.
- `aws.manifestServiceLogExcludeEventTypesum` den Empfang von Protokollen für bestimmte Manifest-Dienstinteraktionen zu beenden.

Eine Liste der Protokoll- und MediaTailor Ereignistypen, die ausgesendet werden, finden Sie unter [Manifest-ProtokolleADS-Protokolle](#), und [Protokolle transkodieren](#).

Wenn Sie keine Abfrageparameter für die Protokollfilterung übergeben, werden alle Protokolle an Ihr Lieferziel MediaTailor geschrieben.

Example serverseitige Sitzungsinitialisierung mit Protokollfiltern

Um PARSING_ERROR Ereignisse aus Ihren Manifestprotokollen und MAKING_ADS_REQUEST aus den ADS-Protokollen auszuschließenGENERATED_MANIFEST, würde die Anfrage zur Sitzungsinitialisierung wie folgt aussehen:

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/index.m3u8?
aws.logMode=DEBUG&aws.manifestServiceLogExcludeEventTypes=GENERATED_MANIFEST,PARSING_ERROR&aws.
```

Um Rohprotokolle aus Ihrem ADS zu aktivieren, geben Sie den RAW_ADS_RESPONSE Wert für den AdsInteractionPublishOptInEventType Parameter an:

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/index.m3u8?
aws.adsInteractionPublishOptInEventType=RAW_ADS_RESPONSE
```

Example clientseitige Sitzungsinitialisierung mit Protokollfiltern

Um Protokollereignisse während der clientseitigen Sitzungsinitialisierung auszuschließen, fügen Sie Parameter in die POST-Anfrage Ihres Clients ein `availSuppression` und protokollieren Sie sie. MediaTailor Weitere Hinweise zum Erstellen einer clientseitigen Wiedergabesitzungsanforderung finden Sie unter [Kundenseitiges Ad-Tracking](#). Im folgenden Beispiel werden PARSING_ERROR Ereignisse aus Ihren Manifestprotokollen CONFIG_SECURITY_ERROR und aus den ADS-Protokollen ausgeschlossen. MAKING_ADS_REQUEST

```
POST parent.m3u8
{
  "adsInteractionLog": {
    ...
    "excludeEventTypes": [
      "MAKING_ADS_REQUEST"
    ]
  },
  "manifestServiceLog": {
    ...
    "excludeEventTypes": [
      "GENERATED_MANIFEST",
      "PARSING_ERROR"
    ]
  }
}
```

```
    ]
  },
  "logMode": "DEBUG"
}
```

Um Rohprotokolle aus Ihrem ADS zu aktivieren, geben Sie den `RAW_ADS_RESPONSE` Wert für den `publishOptInEventTypes` Parameter an:

```
POST parent.m3u8
{
  "adsInteractionLog": {
    "publishOptInEventTypes": ["RAW_ADS_RESPONSE"],
    "excludeEventTypes": [
      "MAKING_ADS_REQUEST"
    ]
  },
  "manifestServiceLog": {
    ...
    "excludeEventTypes": [
      "GENERATED_MANIFEST",
      "PARSING_ERROR"
    ]
  },
  "logMode": "DEBUG"
}
```

Protokollfilter für die Konfiguration pro Wiedergabe

Verwenden Sie die Einstellungen der Wiedergabekonfiguration, um die Protokoll-Ereignistypen zu definieren, die MediaTailor standardmäßig in dieser Wiedergabekonfiguration ausgegeben werden. MediaTailor verwendet diese Standardeinstellungen für die Protokollfilterung für alle Sitzungen, die keine Filterabfrageparameter in der Sitzungsstartanforderung enthalten.

Sie können wählen, ob Sie Folgendes tun möchten:

- Empfangen Sie Protokolle für bestimmte Interaktionen mit dem Ad Decision Server (ADS).
- Schließen Sie Protokolle für bestimmte ADS-Interaktionen aus.
- Schließen Sie Protokolle für bestimmte manifeste Dienstinteraktionen aus.

Informationen zum Einstellen dieser Einstellungen von der MediaTailor Konsole aus finden Sie unter [Konfiguration erstellen](#). Informationen zur MediaTailor API finden Sie unter [PutPlaybackConfiguration](#) in der API-Referenz zu AWS Elemental MediaTailor

Eine Liste der Protokoll- und MediaTailor Ereignistypen, die ausgegeben werden, finden Sie unter [Manifest-ProtokolleADS-Protokolle](#), und [Protokolle transkodieren](#).

AWS Elemental MediaTailor Debug-Protokolle generieren

Verwenden Sie Debug-Protokolle, um Probleme bei der Wiedergabesitzung beim Einfügen von MediaTailor Anzeigen zu beheben. Um Debug-Logs zu generieren, stellen Sie in der Anfrage des Players den Protokollmodus auf Debuggen ein. MediaTailor Stellen Sie für serverseitige Berichte den Protokollmodus in der Wiedergabeanforderung ein. Für clientseitige Berichte legen Sie den Protokollmodus in der Anforderung zur Sitzungsinitialisierung fest.

Wenn der Protokollmodus auf Debug gesetzt ist, werden alle Protokollereignistypen in Logs MediaTailor geschrieben. CloudWatch Die Protokolle enthalten Informationen zu den folgenden Ereignissen. Eine vollständige Liste der in den Debug-Protokollen erzeugten Daten finden Sie unter [Debug-Protokollfelder](#).

- Origin-Interaktion — Details zu MediaTailor Interaktionen mit dem Ursprungsserver. Zum Beispiel die Antwort auf das Ursprungs-Manifest, der Manifesttyp und die Quell-URL.
- Generiertes Manifest — Details zur Antwort auf die Playback-Sitzung von MediaTailor. Zum Beispiel das Manifest, das MediaTailor generiert wird.
- Sitzung initialisiert — Details zur Sitzungsinitialisierung, z. B. die Sitzungs-ID.

Informationen zum Anpassen der Protokollereignistypen, die Sie pro Sitzung erhalten, finden Sie unter [Filterung von Protokollen und Ereignissen](#)

Voraussetzungen

Um den Protokollmodus auf Debug einzustellen, müssen Sie zunächst die MediaTailor Erlaubnis zum Senden von Protokollen an erteilen CloudWatch, falls Sie dies noch nicht getan haben. Sobald Sie die Zugriffsberechtigung MediaTailor erteilt haben CloudWatch, können Sie den Debug-Protokollmodus aktivieren. Informationen zum Erteilen von MediaTailor Zugriffsberechtigungen CloudWatch finden Sie unter [Berechtigungen für Amazon einrichten CloudWatch](#).

Wie stelle ich den Protokollmodus auf Debug ein

In diesem Abschnitt wird erklärt, wie Sie den Protokollmodus für serverseitige und clientseitige Berichte auf Debug einstellen.

Serverseitige Berichterstattung

Fügen Sie für serverseitige Berichte den `?aws.logMode=DEBUG` Abfrageparameter und den Wert in die Wiedergabeanforderung Ihres GET HTTP Players an den HLS- oder DASH-Endpunkt ein. MediaTailor [Allgemeine Informationen zur serverseitigen Berichterstattung finden Sie unter Serverseitige Berichterstattung.](#)

Important

Bei DEBUG-Wert ist die Groß- und Kleinschreibung zu beachten.

Eine Wiedergabeanforderung, die Folgendes beinhaltet, `?aws.logMode=DEBUG` sieht wie folgt aus:

Example Wiedergabeanforderung an einen HLS-Endpunkt

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/<asset-id>?  
aws.logMode=DEBUG
```

Nachdem Sie den Protokollmodus auf Debug gesetzt haben, empfehlen wir Ihnen, zu überprüfen, ob die Debug-Protokollierungssitzung aktiv ist. Um zu überprüfen, ob die Debug-Sitzung aktiv ist, überprüfen Sie, ob CloudWatch Protokolle für die Sitzungs-ID vorhanden sind. Die Sitzungs-ID ist im Wiedergabeendpunkt enthalten, der dies MediaTailor bereitstellt. Weitere Informationen finden Sie unter [Verify that the debug log mode is active for your playback session.](#)

Clientseitige Berichterstattung

Für die clientseitige Berichterstattung fügen Sie den `logMode` Schlüssel und den `DEBUG` Wert in den Text der Anfrage zur POST HTTP Sitzungsinitialisierung Ihres Clients in den `/v1/session`-Endpunkt ein. MediaTailor [Allgemeine Informationen zur clientseitigen Berichterstattung finden Sie unter Clientseitige Berichterstattung.](#)

Important

Bei DEBUG-Wert ist die Groß- und Kleinschreibung zu beachten.

Nachdem Sie den Protokollmodus auf Debug gesetzt haben, empfehlen wir Ihnen, zu überprüfen, ob die Debugsitzung aktiv ist. Um zu überprüfen, ob die Debug-Sitzung aktiv ist, vergewissern Sie sich, dass in den Protokollen ein `SESSION_INITIALIZED` Ereignis mit der Sitzungs-ID verknüpft ist. CloudWatch Die Sitzungs-ID ist im Wiedergabeendpunkt enthalten, der Folgendes MediaTailor bereitstellt. Weitere Informationen finden Sie unter [Verify that the debug log mode is active for your playback session](#).

Maximale Anzahl aktiver Debug-Sitzungen

Sie können maximal 10 aktive Debug-Protokollsitzungen haben. Wenn Ihr Player seine Anfrage zur Sitzungsinitialisierung oder Wiedergabe an sendet MediaTailor, wird MediaTailor überprüft, ob das Limit erreicht wurde. Ist dies der Fall, wird MediaTailor überprüft, ob es veraltete Sessions gibt. Eine Sitzung ist veraltet, wenn innerhalb eines bestimmten Zeitraums nicht darauf zugegriffen wurde. Bei Live-Streams beträgt dieser Zeitraum 10 Minuten, bei VOD-Streams 30 Minuten.

Wenn das maximale Limit für aktive Debug-Protokollsitzungen erreicht wurde, werden Debug-Protokolle für Ihre Sitzung nicht in die CloudWatch Protokolle geschrieben. Wenn Sie in den Protokollen für Ihre Sitzung keine CloudWatch Debug-Protokolle sehen, haben Sie dieses Limit möglicherweise erreicht. Informationen darüber, ob das Limit erreicht wurde, finden Sie unter [Verify that the debug log mode is active for your playback session](#).

Debuggen von Protokollfeldern

In der folgenden Tabelle sind die Debug-Logfelder aufgeführt, in die MediaTailor geschrieben wird. CloudWatch

Feld	Beschreibung
<code>awsAccountId</code>	Ihre AWS-Konto ID.
<code>customerId</code>	Ihre MediaTailor Kundennummer.
<code>eventTimestamp</code>	Der ISO 8601-Zeitstempel, der dem Debug-Log-Ereignis zugeordnet ist.
<code>eventType</code>	Der Typ des Debug-Protokollereignisses. Werte:

Feld	Beschreibung
	• ORIGIN_INTERACTION — Details zu MediaTailor Interaktionen mit dem Ursprungsserver. Zum Beispiel die Antwort auf das Ursprungs-Manifest, der Manifesttyp und die Quell-URL. • GENERATED_MANIFEST — Details zur Antwort auf die Wiedergabebesitzung von MediaTailor. Zum Beispiel das Manifest, das MediaTailor generiert wird. • SESSION_INITIALIZED — Details zur Sitzungsinitialisierung, z. B. die Sitzungs-ID.
<code>originRequestUrl</code>	Die URL Ihres Ursprungsservers, der für diese Anfrage abgerufen wird.
<code>mediaTailorPath</code>	Der MediaTailor Endpunkt, der aufgerufen wurde, einschließlich aller Parameter, die MediaTailor in der ersten Manifestanforderung übergeben wurden.
<code>requestId</code>	Die ID einer bestimmten HTTP-Anfrage an MediaTailor.
<code>responseBody</code>	Das Manifest im Antworttext von MediaTailor. Dies ist entweder das rohe Ursprungsmanifest oder das von generierte Manifest MediaTailor.
<code>sessionId</code>	Die ID der Wiedergabe-Sitzung.
<code>sessionType</code>	Die Art der Wiedergabebesitzung. Werte: HLS, DASH

Lesen Sie die Debug-Protokolle

MediaTailor schreibt die Debug-Logs in Amazon CloudWatch Logs. Es fallen typische CloudWatch Logs-Gebühren an. Verwenden Sie CloudWatch Insights, um die Debug-Logs zu lesen.

Informationen zur Verwendung von CloudWatch Logs Insights finden Sie unter [Analysieren von Protokolldaten mit CloudWatch Logs Insights](#) im AWS CloudWatch Logs-Benutzerhandbuch.

 Note

Es kann einige Minuten dauern, bis die Debug-Protokolle angezeigt werden. CloudWatch Wenn Sie die Protokolle nicht sehen, warten Sie ein paar Minuten und versuchen Sie es erneut. Wenn Sie die Protokolle immer noch nicht sehen, haben Sie möglicherweise die maximale Anzahl aktiver Debug-Protokollsitzungen erreicht. Um zu überprüfen, ob dies der Fall ist, führen Sie eine CloudWatch Abfrage aus, um festzustellen, ob für Ihre Playback-Sitzung eine Debug-Sitzung initialisiert wurde. Weitere Informationen finden Sie unter [Verify that the debug log mode is active for your playback session](#).

Beispiele

Dieser Abschnitt enthält Beispielabfragen, mit denen Sie MediaTailor Debug-Protokolldaten lesen können.

Example 1: Stellen Sie sicher, dass der Debug-Protokollmodus für Ihre Wiedergabesitzung aktiv ist

```
fields @timestamp, @message
| filter sessionId = "32002de2-837c-4e3e-9660-f3075e8dfd90"
| filter eventType = "SESSION_INITIALIZED" # client-side reporting
or mediaTailorPath like "/v1/master" # server-side reporting HLS
or mediaTailorPath like "/v1/dash" # server-side reporting DASH
```

Example 2: Sehen Sie sich die Antworten von Ihrem Absender an

```
fields @timestamp, responseBody, @message, mediaTailorPath
| filter eventType = "ORIGIN_MANIFEST" and sessionId = "32002de2-837c-4e3e-9660-f3075e8dfd90"
```

Example 3: Sehen Sie sich das Manifest an, das von MediaTailor für eine bestimmte Sitzung generiert wurde

```
fields @timestamp, responseBody, @message
| filter mediaTailorPath like "/v1/master/" and eventType = "GENERATED_MANIFEST" and
sessionId = "32002de2-837c-4e3e-9660-f3075e8dfd90"
```

Example 4: Alle Ereignisse für eine bestimmte Zeit anzeigen **requestId**

Verwenden Sie diese Abfrage, um das Ursprungsmanifest und das von generierte Manifest anzuzeigen MediaTailor.

```
fields @timestamp, responseBody, @message, mediaTailorPath
| filter requestId = "e5ba82a5-f8ac-4efb-88a0-55bed21c45b4"
```

Überwachung AWS Elemental MediaTailor mit CloudWatch Amazon-Metriken

Sie können AWS Elemental MediaTailor Metriken überwachen mit CloudWatch. CloudWatch sammelt Rohdaten über die Leistung des Dienstes und verarbeitet diese Daten zu lesbaren Metriken, die nahezu in Echtzeit verfügbar sind. Diese Statistiken werden 15 Monate gespeichert, damit Sie auf Verlaufsinformationen zugreifen können und einen besseren Überblick darüber erhalten, wie Ihre Webanwendung oder der Service ausgeführt werden. Sie können auch Alarme einrichten, die auf bestimmte Grenzwerte achten und Benachrichtigungen senden oder Aktivitäten auslösen, wenn diese Grenzwerte erreicht werden. Weitere Informationen finden Sie im [CloudWatch Amazon-Benutzerhandbuch](#).

Metriken können nützlich sein, wenn Sie veraltete Manifeste untersuchen. Weitere Informationen finden Sie unter [Verwendung von Metriken zur Diagnose veralteter Manifeste](#).

Metriken werden zunächst nach dem Service-Namespace und anschließend nach den verschiedenen Dimensionskombinationen in den einzelnen Namespaces gruppiert.

Um Metriken mit der CloudWatch Konsole anzuzeigen

1. Öffnen Sie die CloudWatch Konsole unter <https://console.aws.amazon.com/cloudwatch/>.
2. Wählen Sie im Navigationsbereich Metriken aus.
3. Wählen Sie unter Alle Metriken den MediaTailorNamespace aus.
4. Wählen Sie zur Ansicht der Metriken die Metrikdimension aus (z. B. originID).
5. Geben Sie den Zeitraum an, den Sie anzeigen möchten.

Um Metriken mit dem AWS Command Line Interface (AWS CLI) anzuzeigen

- Geben Sie in einer Eingabeaufforderung den folgenden Befehl ein:

```
aws cloudwatch list-metrics --namespace "AWS/MediaTailor"
```

AWS Elemental MediaTailor CloudWatch Metriken

Der AWS Elemental MediaTailor Namespace umfasst die folgenden Metriken. Diese Metriken werden standardmäßig in Ihrem Konto veröffentlicht.

Metriken von Channel Assembly (CA)

In der folgenden Tabelle sind alle Metriken nach Kanal oder nach Kanalausgabe verfügbar.

Metrik	Beschreibung
4xxErrorCount	Die Anzahl der 4xx Fehler.
5xxErrorCount	Die Anzahl der 5xx Fehler.
RequestCount	Die Gesamtanzahl der -Anforderungen. Die Anzahl der Transaktionen hängt weitgehend davon ab, wie oft Spieler aktualisierte Manifeste anfordern und wie viele Spieler sie haben. Jede Player-Anforderung zählt als eine Transaktion.
TotalTime	Die Zeit, die der Anwendungsserver für die Bearbeitung der Anfrage benötigte, einschließlich der Zeit, die für den Empfang und das Schreiben von Bytes vom Client und Netzwerk benötigt wurde.

Metriken zum serverseitigen Einfügen von Werbung (SSAI)

In der folgenden Tabelle sind Metriken zum Einfügen von serverseitiger Werbung aufgeführt.

Metrik	Beschreibung
AdDecisionServer.Ads	Die Anzahl der Anzeigen, die in den Antworten des Ad Decision Servers (ADS) innerhalb

Metrik	Beschreibung
	des von Ihnen CloudWatch angegebenen Zeitraums enthalten waren.
<code>AdDecisionServer.Duration</code>	Die Gesamtdauer in Millisekunden aller Anzeigen, die innerhalb des von Ihnen angegebenen CloudWatch Zeitraums vom ADS MediaTailor empfangen wurden. Diese Dauer kann länger als die von Ihnen angegebene Dauer <code>Avail.Duration</code> sein.
<code>AdDecisionServer.Errors</code>	Die Anzahl der Antworten mit 200-Statuscode, die nicht über HTTP erfolgen, der leeren Antworten und der Antworten mit Timeout, die innerhalb des von Ihnen angegebenen CloudWatch Zeitraums vom ADS MediaTailor empfangen wurden.
<code>AdDecisionServer.FillRate</code>	Der einfache Durchschnitt der Raten, mit denen die Antworten vom ADS die entsprechenden einzelnen Ad-Avails für den von Ihnen angegebenen Zeitraum gefüllt haben. Um den gewichteten Durchschnittswert zu erhalten, berechnen Sie die <code>AdDecisionServer.Duration</code> als Prozentsatz der <code>Avail.Duration</code>. Weitere Informationen über einfache und gewichtete Durchschnittswerte finden Sie unter Einfache und gewichtete Durchschnittswerte.
<code>AdDecisionServer.Latency</code>	Die Antwortzeit in Millisekunden für Anfragen an das ADS. MediaTailor
<code>AdDecisionServer.Timeouts</code>	Die Anzahl der Anfragen an das ADS mit Timeout in dem von Ihnen angegebenen CloudWatch Zeitraum.

Metrik	Beschreibung
AdNotReady	Gibt an, wie oft der ADS in dem von Ihnen angegebenen Zeitraum auf eine Anzeige verwiesen hat, die von dem internen Transcode r-Service noch nicht transcodiert worden war. Ein hoher Wert für diese Metrik kann zu einem niedrigen Gesamtwert für Avail.FillRate betragen.
AdsBilled	Die Anzahl der Anzeigen, für die Kunden MediaTailor Rechnungen auf der Grundlage der Insertion in Rechnung gestellt werden.
Avail.Duration	Die geplante Gesamtzahl der Millisekunden, an denen die Anzeige innerhalb des Zeitraums verfügbar ist. CloudWatch Die geplante Gesamtzahl basiert auf der Dauer der Anzeigenverfügbarkeit im Ursprungsmanifest.
Avail.FilledDuration	Die geplante Anzahl von Millisekunden an Werbeanzeigen, die innerhalb des Zeitraums mit MediaTailor Werbeanzeigen gefüllt werden. CloudWatch

Metrik	Beschreibung
<code>Avail.FillRate</code>	Der geplante einfache Durchschnitt der Raten, zu denen einzelne Anzeigen innerhalb des Zeitraums geschaltet MediaTailor werden. CloudWatch Um den gewichteten Durchschnittswert zu erhalten, berechnen Sie die <code>Avail.FilledDuration</code> als Prozentsatz der <code>Avail.Duration</code>. Weitere Informationen über einfache und gewichtete Durchschnittswerte finden Sie unter Einfache und gewichtete Durchschnittswerte. Welchen Wert MediaTailor maximal für <code>Avail.FillRate</code> erzielen kann, ist von dem Wert für <code>AdDecisionServer.FillRate</code> abhängig. Wenn der Wert für <code>Avail.FillRate</code> niedrig ist, vergleichen Sie ihn mit dem Wert für <code>AdDecisionServer.FillRate</code>. Wenn die <code>AdDecisionServer.FillRate</code> niedrig ist, gibt Ihr ADS möglicherweise nicht genügend Werbung für die Avail-Dauer zurück.
<code>Avail.Impression</code>	Die Anzahl der Anzeigen mit Impressions-Tracking-Ereignissen, die beim MediaTailor serverseitigen Beaconsing beobachtet werden (nicht die Anzahl der Impressionen).
<code>Avail.ObservedDuration</code>	Die beobachtete Gesamtzahl der Millisekunden an Anzeigenzugriffen innerhalb des Zeitraums. CloudWatch <code>Avail.ObservedDuration</code> wird am Ende der Werbeanzeige ausgegeben und basiert auf der Dauer der Segmente, über die während der Verfügbarkeit der Anzeige im Manifest berichtet wurde.

Metrik	Beschreibung
Avail.ObservedFilledDuration	Die beobachtete Anzahl von Millisekunden der Werbeanzeige, die innerhalb des Zeitraums mit Werbeanzeigen MediaTailor gefüllt wurde. CloudWatch
Avail.ObservedFillRate	Der beobachtete einfache Durchschnitt der Raten, MediaTailor mit denen gefüllte einzelne Anzeigen innerhalb des Zeitraums geschaltet wurden. CloudWatch Wird nur für HLS-Manifeste beim ersten CUE-IN Tag ausgegeben. Wenn kein CUE-IN Tag vorhanden ist, wird diese Metrik MediaTailor nicht ausgegeben.
Avail.ObservedSlateDuration	Die beobachtete Gesamtzahl der Millisekunden von Schiefer, die innerhalb des Zeitraums eingefügt wurden. CloudWatch
GetManifest.Age	Das Gesamtalter des Manifests in Millisekunden. Gemessen von dem Zeitpunkt, zu dem der Ursprung das Manifest erstellt, bis zu dem Zeitpunkt, zu dem das personalisierte Manifest MediaTailor gesendet wird. Weitere Informationen zu Metriken zur Messung des Alters von Manifesten finden Sie unter Verwendung von Metriken zur Diagnose veralteter Manifeste.
GetManifest.Errors	Die Anzahl der Fehler, die MediaTailor beim Generieren von Manifesten in dem von Ihnen angegebenen CloudWatch Zeitraum aufgetreten sind.

Metrik	Beschreibung
GetManifest.Latency	Die MediaTailor Antwortzeit in Millisekunden für die Anforderung zur Generierung von Manifesten. Weitere Informationen zu Metriken zur Messung des Alters von Manifesten finden Sie unter. Verwendung von Metriken zur Diagnose veralteter Manifeste
GetManifest.MediaTailorAge	Die Zeitspanne in Millisekunden, MediaTailor in der das Manifest gespeichert wurde. Gemessen vom MediaTailor Empfang einer ursprünglichen Antwort bis zum Versand des personalisierten MediaTailor Manifests. Weitere Informationen zu Metriken zur Messung des Alters von Manifesten finden Sie unter Verwendung von Metriken zur Diagnose veralteter Manifeste.
Origin.Age	Die Zeitspanne in Millisekunden, für die der Ursprung das Manifest hat. Gemessen von dem Zeitpunkt, zu dem der Ursprung das Manifest erstellt, bis zu dem Zeitpunkt, an dem die ursprüngliche MediaTailor Anfrage gesendet wird. Alle <code>origin.*</code> Metriken werden für Anfragen ausgegeben, die direkt vom Ursprung aus erfüllt werden. Sie werden nicht für zwischeng gespeicherte Ausgangsantworten ausgegeben. Weitere Informationen zu Metriken zur Messung des Manifestalters finden Sie unter Verwendung von Metriken zur Diagnose veralteter Manifeste.

Metrik	Beschreibung
<code>Origin.Errors</code>	Die Anzahl der 200-Statuscode-Antworten ohne HTTP-Status und der Antworten mit Timeout, die in dem von Ihnen angegebenen CloudWatch Zeitraum vom Ursprungsserver MediaTailor empfangen wurden. Alle <code>origin.*</code> Messwerte werden für Anfragen ausgegeben, die direkt vom Ursprung aus erfüllt werden. Sie werden nicht für zwischeng espeicherte Ausgangsantworten ausgegeben.
<code>Origin.ManifestFileSizeBytes</code>	Die Dateigröße des Ursprungsmanifests in Byte für HLS und DASH. In der Regel wird diese Metrik in Verbindung mit <code>Origin.ManifestFileSizeTooLarge</code> verwendet. Alle <code>origin.*</code> Metriken werden für Anfragen ausgegeben, die direkt vom Ursprung aus erfüllt werden. Sie werden nicht für zwischeng espeicherte Ausgangsantworten ausgegeben.
<code>Origin.ManifestFileSizeTooLarge</code>	Die Anzahl der Antworten vom Ursprung, deren Manifestgröße die konfigurierte Menge übersteigt. In der Regel wird diese Metrik in Verbindung mit <code>Origin.ManifestFileSizeBytes</code> verwendet. Alle <code>origin.*</code> Metriken werden für Anfragen ausgegeben, die direkt vom Ursprung aus erfüllt werden. Sie werden nicht für zwischeng espeicherte Ausgangsantworten ausgegeben.

Metrik	Beschreibung
<code>Origin.Timeouts</code>	Die Anzahl der Anfragen an den Ursprungs server, bei denen das Timeout in dem von Ihnen angegebenen CloudWatch Zeitraum überschritten wurde. Alle <code>origin.*</code> Metriken werden für Anfragen ausgegeben, die direkt vom Ursprung aus erfüllt werden. Sie werden nicht für zwischeng espeicherte Ausgangsantworten ausgegeben.
<code>Requests</code>	Die Anzahl gleichzeitiger Transaktionen pro Sekunde für alle Anforderungstypen. Die Anzahl der Transaktionen hängt hauptsäch lich von der Anzahl der Spieler ab und davon, wie oft die Spieler aktualisierte Manifeste anfordern. Jede Player-Anforderung zählt als eine Transaktion.
<code>SkippedReason.DurationExceeded</code>	Die Anzahl der Anzeigen, die nicht in eine Avail eingefügt wurden, weil die ADS eine Anzeigend auer zurückgab, die über der angegebenen Verfügbarkeitsdauer lag. Ein hoher Wert für diese Kennzahl kann zu einer Diskrepanz zwischen der Metrik und beitragen. <code>Avail.Ads</code> <code>AdDecisionServer.Ads</code>
<code>SkippedReason.EarlyCueIn</code>	Die Anzahl der Anzeigen, die aufgrund eines vorzeitigen Fehlers übersprungen wurden. CUE-IN
<code>SkippedReason.ImportError</code>	Die Anzahl der Anzeigen, die aufgrund eines Fehlers im Importjob übersprungen wurden.
<code>SkippedReason.ImportInProgress</code>	Die Anzahl der Anzeigen, die aufgrund eines vorhandenen aktiven Importauftrags übersprun gen wurden.

Metrik	Beschreibung
<code>SkippedReason.InternalError</code>	Die Anzahl der Anzeigen, die aufgrund eines MediaTailor internen Fehlers übersprungen wurden.
<code>SkippedReason.NewCreative</code>	Die Anzahl der Anzeigen, die nicht in ein Avail eingefügt wurden, weil das Asset zum ersten Mal von einem Kunden angefordert wurde. Ein hoher Wert für diese Kennzahl kann vorübergehend zu einem niedrigen Gesamtwert <code>Avail.FillRate</code> führen, bis die Assets erfolgreich transkodiert werden können.
<code>SkippedReason.NoVariantMatch</code>	Die Anzahl der Anzeigen, die übersprungen wurden, weil es keine Variantenübereinstimmung zwischen der Anzeige und dem Inhalt gab.
<code>SkippedReason.PersonalizationThresholdExceeded</code>	Die Dauer von Anzeigen, die in dieser Konfiguration den für die Personalisierung festgelegten Schwellenwert überschreiten.
<code>SkippedReason.ProfileNotFound</code>	Die Anzahl der Anzeigen, die übersprungen wurden, weil das Transcodierungsprofil nicht gefunden wurde.
<code>SkippedReason.TranscodeError</code>	Die Anzahl der Anzeigen, die aufgrund eines Transcodierungsfehlers übersprungen wurden.
<code>SkippedReason.TranscodeInProgress</code>	Die Anzahl der Anzeigen, die nicht in einen Avail eingefügt wurden, weil die Anzeige noch nicht transkodiert wurde. Ein hoher Wert für diese Kennzahl kann vorübergehend zu einem niedrigen Gesamtwert <code>Avail.FillRate</code> führen, bis die Inhalte erfolgreich transkodiert werden können.

Einfache und gewichtete Durchschnittswerte

Sie können den einfachen Durchschnitt und den gewichteten Durchschnitt für die Antworten des ADS auf Anzeigenanfragen von MediaTailor und für die Anzahl der MediaTailor verfügbaren Anzeigen abrufen:

- Die einfachen Durchschnittswerte werden in `AdDecisionServer.FillRate` und `Avail.FillRate` angegeben. Dabei handelt es sich um die Durchschnittswerte der Füllraten in Prozent für einzelne Avails für den Zeitraum. Bei den einfachen Durchschnittswerten werden keine Unterschiede in der Dauer der einzelnen Avails berücksichtigt.
- Die gewichteten Durchschnittswerte beziehen sich auf die prozentuale Füllrate für die Summe aller Werte der Avail-Dauer. Diese Werte werden als $(\text{AdDecisionServer.Duration} * 100) / \text{Avail.Duration}$ und $(\text{Avail.FilledDuration} * 100) / \text{Avail.Duration}$ berechnet. Diese Durchschnittswerte reflektieren die Unterschiede in der Dauer der einzelnen Ad-Avails, sodass solche mit längerer Dauer stärker gewichtet werden.

Für einen Zeitraum mit nur einem einzigen Ad-Avail entspricht der mit `AdDecisionServer.FillRate` angegebene einfache Durchschnitt dem mit $(\text{AdDecisionServer.Duration} * 100) / \text{Avail.Duration}$ angegebenen gewichteten Durchschnitt. Der mit `Avail.FillRate` angegebene einfache Durchschnitt entspricht dem mit $(\text{Avail.FilledDuration} * 100) / \text{Avail.Duration}$ angegebenen gewichteten Durchschnitt.

Beispiel

Angenommen, die von Ihnen angegebene Zeitspanne verfügt über die folgenden zwei Ad-Avails:

- Die Dauer des ersten Ad-Avail beträgt 90 Sekunden:
 - Die ADS-Antwort für den Avail stellt 45 Sekunden an Werbung bereit (zu 50 % gefüllt).
 - MediaTailor füllt 45 Sekunden der verfügbaren Anzeigenzeit aus (50% gefüllt).
- Die Dauer des zweiten Ad-Avail beträgt 120 Sekunden:
 - Die ADS-Antwort für den Avail stellt 120 Sekunden an Werbung bereit (zu 100 % gefüllt).
 - MediaTailor füllt 90 Sekunden der verfügbaren Werbezeit aus (75% gefüllt).

Die Metriken sind wie folgt:

- Der Wert von `Avail.Duration` ist 120, nämlich die Summe der Dauer der beiden Ad-Avail-Zeiträume: $90 + 120$.
- `AdDecisionServer.Duration` ist 165, nämlich die Summe der beiden Werte der Antwortdauer: $45 + 120$.
- `Avail.FilledDuration` ist 135, nämlich die Summe der beiden Werte für die gefüllte Dauer: $45 + 90$.
- Der Wert von `AdDecisionServer.FillRate` ist 75 %, nämlich der gefüllte prozentuale Durchschnitt für jeden Avail: $(50 \% + 100 \%) / 2$. Dies ist der einfache Durchschnittswert.
- Der gewichtete Durchschnittswert für die ADS-Füllraten ist 78,57 %, was dem Wert für `AdDecisionServer.Duration` als Prozentsatz des Wertes für `Avail.Duration` entspricht: $(165 * 100) / 210$. Diese Berechnung berücksichtigt die Unterschiede bezüglich der Dauer.
- Der Wert von `Avail.FillRate` ist 62.5 %, nämlich der gefüllte prozentuale Durchschnitt für jeden Avail: $(50 \% + 75 \%) / 2$. Dies ist der einfache Durchschnittswert.
- Der gewichtete Durchschnitt der MediaTailor verfügbaren Füllraten liegt bei 64,29%, was einem Prozentsatz der folgenden `Avail.FilledDuration` Werte entspricht `Avail.Duration`: $(135 * 100) / 210$. Diese Berechnung berücksichtigt die Unterschiede bezüglich der Dauer.

Der höchste Wert `Avail.FillRate`, der für jede MediaTailor verfügbare Anzeige erreicht werden kann, liegt bei 100%. Der ADS gibt möglicherweise mehr Werbezeit zurück, als im Avail verfügbar ist, MediaTailor kann aber nur den verfügbaren Zeitraum füllen.

AWS Elemental MediaTailor CloudWatch Abmessungen

Sie können die AWS Elemental MediaTailor Daten anhand der folgenden Dimension filtern.

Dimension	Beschreibung
Configuration Name	Gibt die Konfiguration an, zu der die Metrik gehört.

Verwendung von Metriken zur Diagnose veralteter Manifeste von AWS Elemental MediaTailor

Ein veraltetes Manifest ist ein Manifest, das in letzter Zeit nicht aktualisiert wurde. Bei verschiedenen Workflows zum Einfügen von Anzeigen kann es aufgrund einer Vielzahl von Faktoren (z. B.

Anforderungen an nachgelagerte Systeme) unterschiedlich sein, wie lange es dauern muss, bis ein Manifest als veraltet angesehen wird. Sie können CloudWatch Amazon-Metriken verwenden, um Manifeste zu identifizieren, die die Veraltungstoleranz für Ihren Workflow überschreiten, und dabei helfen, herauszufinden, was die Ursache für die Verzögerungen bei der Aktualisierung von Manifesten sein könnte.

Die folgenden Kennzahlen helfen dabei, veraltete Manifeste und ihre Ursachen zu identifizieren.

Informationen zu allen ausgegebenen Metriken MediaTailor finden Sie unter [AWS Elemental MediaTailor CloudWatch Metriken](#)

Metrik	Definition	Verwenden Sie
GetManifest.Age	Misst das Gesamtalter des Manifests, einschließlich GetManifest.MediaTailorAge sowohl als auch Origin.Age für diese Konfiguration.	Sie können diese Metrik verwenden, um Manifeste zu identifizieren, die Ihren Aktualisierungsschwellenwert überschritten haben und veraltet sind. Stellen Sie Alarmer für diese Metrik ein, sodass Sie benachrichtigt werden, wenn veraltete Manifeste zugestellt werden. Informationen zu Alarmen finden Sie unter Alarming on metrics im CloudWatch Amazon-Benutzerhandbuch. Wenn Sie einen Alarm erhalten, verwenden Sie Origin.Age und, GetManifest.MediaTailorAge um festzustellen, ob MediaTailor oder der Ursprung für die Veralterung verantwortlich ist.
Origin.Age	Misst, wie lange der Ursprung über das Manifest verfügt, bevor es an MediaTailor diese Konfiguration gesendet wird. Diese Metrik wird nicht ausgegeben, wenn die Antwort von einem Content	Wenn Sie veraltete Manifeste mit identifizierenGetManifest.Age, können Sie die Origin.Age Metrik und die GetManifest.MediaTailorAge Metrik analysieren, um festzustellen, was zur manifesten Veralterung beiträgt.

Metrik	Definition	Verwenden Sie
	Delivery Network (CDN) kommt. Die Antwort muss vom Ursprung kommen, damit <code>Origin.Age</code> sie gesendet werden kann.	Wenn Sie feststellen, dass dies länger <code>Origin.Age</code> ist als Ihre typischen Verarbeitungszeiten am Ursprung, ist es wahrscheinlich, dass das Problem vom vorgelagerten System verursacht wird, und Sie sollten die Diagnose darauf konzentrieren.
<code>GetManifest.MediaTailorAge</code>	Misst, wie MediaTailor lange dieses Manifest für diese Konfiguration gespeichert wurde.	Wenn Sie veraltete Manifeste mit identifizieren <code>GetManifest.Age</code>, können Sie die <code>GetManifest.MediaTailorAge</code> Metrik und die <code>Origin.Age</code> Metrik analysieren, um festzustellen, was zur manifesten Veralterung beiträgt. Wenn das länger <code>GetManifest.MediaTailorAge</code> dauert als die typische Personalisierung von Manifesten, ist es wahrscheinlich MediaTailor, dass das Problem dadurch verursacht MediaTailor wird, und Sie sollten die Diagnose darauf konzentrieren. <code>GetManifest.Latency</code> kann außerdem ermitteln, wie lange es dauert, ein personalisiertes Manifest MediaTailor zu erstellen.

Metrik	Definition	Verwenden Sie
GetManifest.Latency	Misst die Zeit, die benötigt wird MediaTailor , um die Anfrage zu bearbeiten und ein personalisiertes Manifest für diese Konfiguration zu erstellen.	Wenn Sie vergleichen <code>Origin.Age</code> <code>GetManifest.MediaTailorAge</code> und feststellen, dass dies die Ursache für die verzögerte Manifestzustellung MediaTailor ist, können Sie die <code>GetManifest.Latency</code> Metrik analysieren, um festzustellen, ob der Prozess der Manifest-Personalisierung zur Veralterung des Manifests beiträgt. <code>GetManifest.MediaTailorAge</code> misst die Gesamtzeit, in der das Manifest gespeichert ist. MediaTailor <code>GetManifest.Latency</code> misst, wie viel von dieser Speicherzeit auf die MediaTailor Personalisierung des Manifests als Antwort auf eine Anfrage entfällt.

AWS Elemental MediaTailor API-Aufrufe aufzeichnen

AWS Elemental MediaTailor ist in einen Dienst integriert AWS CloudTrail, der eine Aufzeichnung der Aktionen bereitstellt, die ein Benutzer, eine Rolle oder ein AWS Dienst in ausgeführt hat MediaTailor. CloudTrail erfasst alle API-Aufrufe MediaTailor als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der MediaTailor Konsole und Codeaufrufen für die MediaTailor API-Operationen. Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Bereitstellung von CloudTrail Ereignissen an einen Amazon S3 S3-Bucket aktivieren, einschließlich Ereignissen für MediaTailor. Wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse trotzdem in der CloudTrail Konsole im Ereignisverlauf anzeigen. Anhand der von gesammelten Informationen können Sie die Anfrage ermitteln CloudTrail, an die die Anfrage gestellt wurde MediaTailor, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere Details.

Weitere Informationen CloudTrail dazu finden Sie im [AWS CloudTrail Benutzerhandbuch](#).

AWS Elemental MediaTailor Informationen in CloudTrail

CloudTrail ist in Ihrem AWS Konto aktiviert, wenn Sie das Konto erstellen. Wenn eine Aktivität in stattfindet AWS Elemental MediaTailor, wird diese Aktivität zusammen mit anderen CloudTrail AWS Serviceereignissen im Ereignisverlauf in einem Ereignis aufgezeichnet. Sie können aktuelle Ereignisse in Ihrem AWS Konto ansehen, suchen und herunterladen. Weitere Informationen finden Sie unter [Ereignisse mit CloudTrail Ereignisverlauf anzeigen](#).

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem AWS Konto, einschließlich der Ereignisse für AWS Elemental MediaTailor, erstellen Sie einen Trail. Ein Trail ermöglicht CloudTrail die Übermittlung von Protokolldateien an einen Amazon S3 S3-Bucket. Wenn Sie einen Trail in der Konsole erstellen, gilt der Trail standardmäßig für alle AWS Regionen. Der Trail protokolliert Ereignisse aus allen Regionen der AWS Partition und übermittelt die Protokolldateien an den von Ihnen angegebenen Amazon S3 S3-Bucket. Darüber hinaus können Sie andere AWS Dienste konfigurieren, um die in den CloudTrail Protokollen gesammelten Ereignisdaten weiter zu analysieren und darauf zu reagieren. Weitere Informationen finden Sie hier:

- [Einen Trail für dein AWS Konto erstellen](#)
- [AWS Serviceintegrationen mit Protokollen CloudTrail](#)
- [Konfiguration von Amazon SNS SNS-Benachrichtigungen für CloudTrail](#)
- [Empfangen von CloudTrail Protokolldateien aus mehreren Regionen](#) und [Empfangen von CloudTrail Protokolldateien von mehreren Konten](#)

Alle AWS Elemental MediaTailor Aktionen werden von der [AWS Elemental MediaTailor API-Referenz](#) protokolliert CloudTrail und sind in dieser dokumentiert. Zum Beispiel werden durch Aufrufe der PutPlaybackConfiguration- und ListPlaybackConfigurations-Operationen Einträge in den CloudTrail-Protokolldateien generiert.

Jeder Ereignis- oder Protokolleintrag enthält Informationen zu dem Benutzer, der die Anforderung generiert hat. Die Identitätsinformationen unterstützen Sie bei der Ermittlung der folgenden Punkte:

- Ob die Anfrage mit dem Root-Benutzer oder AWS Identity and Access Management (IAM) -Anmeldeinformationen gestellt wurde
- Ob die Anfrage mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen föderierten Benutzer ausgeführt wurde
- Ob die Anfrage von einem anderen AWS Dienst gestellt wurde

Weitere Informationen finden Sie unter [CloudTrail -Element userIdentity](#).

Grundlagen zu AWS Elemental MediaTailor -Protokolldateieinträgen

Ein Trail ist eine Konfiguration, die die Übertragung von Ereignissen als Protokolldateien an einen von Ihnen angegebenen Amazon S3 S3-Bucket ermöglicht. CloudTrail Protokolldateien enthalten einen oder mehrere Protokolleinträge. Ein Ereignis ist eine einzelne Anforderung aus einer beliebigen Quelle und enthält Informationen zur angeforderten Aktion, zu Datum und Uhrzeit der Aktion, zu den Anforderungsparametern usw. CloudTrail -Protokolldateien stellen kein geordnetes Stack-Trace der öffentlichen API-Aufrufe dar. Daher werden sie nicht in einer bestimmten Reihenfolge angezeigt.

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die PutPlaybackConfiguration Aktion demonstriert:

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDAEXAMPLE",
    "arn": "arn:aws:iam::111122223333:user/testuser",
    "accountId": "111122223333",
    "accessKeyId": "AIDAEXAMPLE",
    "userName": "testuser"
  },
  "eventTime": "2018-12-28T22:53:46Z",
  "eventSource": "mediatailor.amazonaws.com",
  "eventName": "PutPlaybackConfiguration",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "1.2.3.4",
  "userAgent": "PostmanRuntime/7.4.0",
  "requestParameters": {
    "VideoContentSourceUrl": "http://examplevideo.com",
    "Name": "examplename",
    "AdDecisionServerUrl": "http://exampleads.com"
  },
  "responseElements": {
    "SessionInitializationEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/session/AKIAIOSFODNN7EXAMPLE/examplename/",
    "DashConfiguration": {
      "ManifestEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/dash/AKIAIOSFODNN7EXAMPLE/examplename/",
      "MpdLocation": "EMT_DEFAULT"
    }
  }
}
```

```

    },
    "AdDecisionServerUrl": "http://exampleads.com",
    "CdnConfiguration": {},
    "PlaybackEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com",
    "HlsConfiguration": {
      "ManifestEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/master/AKIAIOSFODNN7EXAMPLE/exemplename/"
    },
    "VideoContentSourceUrl": "http://examplevideo.com",
    "Name": "exemplename"
  },
  "requestID": "1a2b3c4d-1234-5678-1234-1a2b3c4d5e6f",
  "eventID": "987abc65-1a2b-3c4d-5d6e-987abc654def",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
}

```

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die `GetPlaybackConfiguration` Aktion demonstriert:

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDAEXAMPLE",
    "arn": "arn:aws:iam::111122223333:user/testuser",
    "accountId": "111122223333",
    "accessKeyId": "AIDAEXAMPLE",
    "userName": "testuser"
  },
  "eventTime": "2018-12-28T22:52:37Z",
  "eventSource": "mediatailor.amazonaws.com",
  "eventName": "GetPlaybackConfiguration",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "1.2.3.4",
  "userAgent": "PostmanRuntime/7.4.0",
  "requestParameters": {
    "Name": "exemplename"
  },
  "responseElements": null,
  "requestID": "0z1y2x3w-0123-4567-9876-6q7r8s9t0u1v",
}

```

```

    "eventID": "888ddd77-3322-eeww-uuii-abc123jkl343",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
  }

```

Empfangen von AWS Elemental MediaTailor Channel Assembly-Benachrichtigungen

MediaTailor erstellt Warnmeldungen für Probleme oder potenzielle Probleme, die mit Ihren Ressourcen für die Kanalmontage auftreten. Die Warnung beschreibt das Problem, wann das Problem aufgetreten ist und welche Ressourcen betroffen sind.

Sie können die Warnungen in der AWS Management Console, AWS Command Line Interface (AWS CLI) oder programmgesteuert mithilfe der MediaTailor [ListAlerts](#) API anzeigen. AWS SDKs

Important

Benachrichtigungen sind nur für Channel-Assembly-Ressourcen verfügbar, die am oder nach dem 14. Juli 2021 erstellt wurden.

Benachrichtigungen zur Kanalmontage

Art der Warnung	Warnungsc ode	Warnmeldung	Hinweise
VOD-Quelle	NOT_PROCESSED	MediaTailor hat die Paketkonfiguration <i>configurationPath</i> nicht verarbeitet.	
	UNREACHABLE	Wir können die URL nicht erreichen <i>url</i> .	
	UNAUTHORIZED	<i>url</i> hat die Anfrage nicht autorisiert.	

Art der Warnung	Warnungscode	Warnmeldung	Hinweise
	TIMEOUT	Bei der Verbindung mit <i>url</i> trat eine Zeitüberschreitung auf.	
	UNPARSABLE_MANIFEST	MediaTailor ist beim Analysieren des Manifests von auf ein Problem gestoßen. <i>url</i>	
	VARIANT_DURATION_MISMATCH	MediaTailor ist beim Parsen des Manifests von auf Varianten gestoßen, deren Gesamtdauer nicht übereinstimmt. <i>url</i> Dies kann zu Verzögerungen bei der Wiedergabe führen.	Ihr Manifest hat je nach Varianten/Repräsentationen unterschiedliche Dauern. Dies kann dazu führen, dass Bildunterschriften fehlen oder falsch sind und Anzeigen nicht eingefügt MediaTailor werden können.
	SEGMENT_DURATION_TOO_LONG	MediaTailor ist beim Analysieren des Manifests von auf ein Segment mit einer Dauer von mehr als dreißig Sekunden gestoßen. <i>url</i> Dies kann zu Verzögerungen bei der Wiedergabe, fehlenden oder falschen Untertiteln und der Unfähigkeit, Werbung einzufügen, führen.	Ihr Manifest enthält ein Segment, das länger als 30 Sekunden ist.

Art der Warnung	Warnungscode	Warnmeldung	Hinweise
	TARGET_DURATION_MISMATCH	MediaTailor ist beim Parsen des Manifests von auf eine Nichtübereinstimmung der EXT-X-TARGETDURATION Werte in den HLS-Manifesten gestoßen. <i>url</i> Dies kann zu Verzögerungen bei der Wiedergabe führen.	Die Zieldauer stimmt nicht für alle Manifeste in der Quelle überein.
Standort der Quelle	NOT_PROCESSED	MediaTailor hat die Ressource nicht verarbeitet <i>resourceName</i> .	
Programm	VOD_SOURCE_ALERT	Die VOD-Quelle <i>vodSourceName</i> in diesem Programm hat die folgende Warnung: <i>vodSourceAlertCode</i> <i>vodSourceAlertMessage</i>	
	SOURCE_LOCATION_ALERT	Für den in diesem Programm <i>sourceLocationName</i> enthaltenen Quellpfad wurde die folgende Warnung angezeigt: <i>sourceLocationAlertCode</i> <i>sourceLocationAlertMessage</i>	

Art der Warnung	Warnungscode	Warnmeldung	Hinweise
	CODEC_MISMATCH	MediaTailor Im <i>channelName</i> Zeitplan ist ein nicht übereinstimmender Codec aufgetreten. Die Nichtübereinstimmung liegt <i>sourceGroupName</i> zwischen dem Manifest <i>manifestUrl</i> und <i>programName1</i> <i>programName2</i> dem Manifest. <i>manifestUrl</i>	
	RESOLUTION_MISMATCH	MediaTailor ist auf eine nicht übereinstimmende Lösung im <i>channelName</i> Zeitplan gestoßen. Die Nichtübereinstimmung liegt <i>sourceGroupName</i> zwischen <i>programName1</i> dem Manifest <i>manifestUrl</i> und <i>programName2</i> dem Manifest. <i>manifestUrl</i>	
	BANDWIDTH_MISMATCH	MediaTailor ist im Zeitplan auf eine nicht übereinstimmende Bandbreite gestoßen. <i>channelName</i> Die Nichtübereinstimmung liegt <i>sourceGroupName</i> zwischen <i>programName1</i> dem Manifest <i>manifestUrl</i> und <i>programName2</i> dem Manifest. <i>manifestUrl</i>	

Art der Warnung	Warnungscode	Warnmeldung	Hinweise
	FRAMERATE_MISMATCH	MediaTailor ist im Zeitplan auf eine nicht übereinstimmende Framerate gestoßen. <i>channelName</i> Die Nichtübereinstimmung liegt <i>sourceGroup</i> zwischen dem <i>programName1</i> Manifest und dem Manifest<i>manifestUrl</i> . <i>programName2</i> <i>manifestUrl</i>	
	TARGET_DURATION_MISMATCH	MediaTailor Es wurden nicht übereinstimmende EXT-X-TARGETDURATION Werte für alle HLS-Manifestdateien im Zeitplan festgestellt. <i>channelName</i> Die Nichtübereinstimmung liegt <i>sourceGroup</i> zwischen dem Manifest <i>manifestUrl</i> und <i>programName1</i> <i>programName2</i> dem Manifest. <i>manifestUrl</i>	

Art der Warnung	Warnungscode	Warnmeldung	Hinweise
	SEGMENT_DURATION_MISMATCH	MediaTailor Es wurde festgestellt, dass die Werte für die Segmentdauer der einzelnen Manifeste im <i>channelName</i> Zeitplan nicht übereinstimmten. Die Nichtübereinstimmung liegt <i>sourceGroupName</i> zwischen <i>programName1</i> dem Manifest <i>manifestUrl</i> und <i>programName2</i> dem Manifest. <i>manifestUrl</i>	
	NO_COMMON_SEGMENT_BOUNDARY_FOR_AD_SLATE	MediaTailor Ich war nicht in der Lage, ein Ad Slate am Offset <i>offsetMillis</i> für das Programm <i>programName</i> einzufügen. Bei der Startzeit des Ad Slate gibt es keine gemeinsame Segmentgrenze.	
	NOT_PROCESSED	MediaTailor hat die Ressource nicht verarbeitet <i>resourceName</i> .	
	TOO_MANY_ALERTS	MediaTailor hat zu viele Alerts gefunden und gibt keine weiteren Alerts für <i>programName</i> . Löschen Sie bestehende Benachrichtigungen , für die Sie weiterhin Benachrichtigungen erhalten möchten <i>programName</i> .	

Art der Warnung	Warnungsc ode	Warnmeldung	Hinweise
Kanal	PROGRAM_A LERT	Das in diesem Kanal <i>programName</i> enthaltene Programm hat die folgende Warnung <i>programAl ertCode : programAl ertMessage</i>	

Warnmeldungen anzeigen

Sie können Benachrichtigungen für jede Ressource zur MediaTailor Kanalmontage anzeigen. Wenn Sie sich die Benachrichtigungen für Kanäle und Programme ansehen, werden MediaTailor auch alle zugehörigen Ressourcen angezeigt, die im Kanal oder Programm enthalten sind. Wenn Sie sich beispielsweise die Benachrichtigungen für ein bestimmtes Programm ansehen, werden Ihnen auch Benachrichtigungen für den Quellstandort und die VOD-Quellen angezeigt, die das Programm enthält.

Gehen Sie wie folgt vor, um Benachrichtigungen anzuzeigen.

Console

Um Benachrichtigungen in der Konsole anzuzeigen

1. Öffnen Sie die MediaTailor Konsole unter <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie die Ressource aus, für die Sie Benachrichtigungen anzeigen möchten.
3. Wählen Sie die Registerkarte „Benachrichtigungen“, um die Benachrichtigungen anzuzeigen.

AWS Command Line Interface (AWS CLI)

Um Benachrichtigungen für eine Channel-Assembly-Ressource aufzulisten, benötigen Sie den [Amazon-Ressourcennamen \(ARN\)](#) der Ressource. Sie können den `describe-resource_type` Befehl in der AWS Command Line Interface (AWS CLI) verwenden, um den ARN der Ressource abzurufen. Führen Sie beispielsweise den Befehl [describe-channel](#) aus, um den ARN eines bestimmten Kanals abzurufen:

```
aws mediatailor describe-channel --channel-name MyChannelName
```

Verwenden Sie dann den Befehl [aws mediatailor list-alerts](#), um die mit der Ressource verknüpften [Warnungen](#) aufzulisten:

```
aws mediatailor list-alerts --resource-arn arn:aws:mediatailor:region:aws-account-id:resource-type/resource-name
```

API

Um Benachrichtigungen für eine Channel-Assembly-Ressource aufzulisten, benötigen Sie den [Amazon-Ressourcennamen \(ARN\)](#) der Ressource. Sie können den `DescribeResource` Vorgang in der MediaTailor API verwenden, um den ARN der Ressource abzurufen. Verwenden Sie den [DescribeChannel](#) Vorgang beispielsweise, um den ARN eines bestimmten Kanals abzurufen.

Verwenden Sie dann die [ListAlerts](#) API, um die Benachrichtigungen für die Ressource aufzulisten.

Umgang mit Warnmeldungen

Wenn eine Warnung auftritt, sehen Sie sich die Warnungen in der AWS Management Console Alerts API an oder verwenden Sie die AWS Command Line Interface (AWS CLI) AWS SDKs, oder die MediaTailor Alerts API, um die möglichen Ursachen des Problems zu ermitteln.

MediaTailor Löscht die Warnung, nachdem Sie das Problem behoben haben.

Ressourcen taggen AWS Elemental MediaTailor

Ein Tag ist eine Metadaten-Bezeichnung, die Sie einer Ressource zuweisen oder die sie einer AWS AWS Ressource zuweist. Jedes Tag besteht aus einem Schlüssel und einem Wert. Für Tags, die Sie zuweisen, definieren Sie einen Schlüssel und einen Wert. So können Sie beispielsweise den Schlüssel als `stage` und den Wert für eine Ressource als `test` definieren.

Tags sind für folgende Aktivitäten nützlich:

- Identifizieren und organisieren Sie Ihre AWS Ressourcen. Viele AWS Dienste unterstützen Tagging, sodass Sie Ressourcen aus verschiedenen Diensten dasselbe Tag zuweisen können, um

anzuzeigen, dass die Ressourcen miteinander verknüpft sind. Sie könnten beispielsweise einem AWS Elemental MediaPackage Kanal und Endpunkt dasselbe Tag zuweisen, das Sie einer AWS Elemental MediaTailor Konfiguration zuweisen.

- Verfolgen Sie Ihre AWS Kosten. Sie aktivieren diese Tags auf dem AWS Fakturierung und Kostenmanagement Dashboard. AWS verwendet die Tags, um Ihre Kosten zu kategorisieren und Ihnen einen monatlichen Kostenverteilungsbericht zu senden. Weitere Informationen finden Sie unter [Use cost allocation tags](#) (Verwendung von Kostenzuordnungs-Tags) im [AWS Billing - Benutzerhandbuch](#).
- Kontrollieren Sie den Zugriff auf Ihre AWS Ressourcen. Weitere Informationen finden Sie unter [Controlling access using tags](#) (Zugriffssteuerung mit Tags) im [IAM-Benutzerhandbuch](#).

In den folgenden Abschnitten finden Sie weitere Informationen zu Tags für AWS Elemental MediaTailor.

Unterstützte Ressourcen in AWS Elemental MediaTailor

Die folgenden Ressourcen in AWS Elemental MediaTailor unterstützen Tagging:

- Kanäle
- Konfigurationen
- SourceLocations
- VodSources

Tag-Einschränkungen

Die folgenden grundlegenden Einschränkungen gelten für Tags auf AWS Elemental MediaTailor Ressourcen:

- Die maximale Anzahl der Tags, die Sie einer Ressource zuweisen können – 50.
- Maximale Schlüssellänge – 128 Unicode-Zeichen.
- Maximale Wertlänge – 256 Unicode-Zeichen.
- Valid characters for key and value – a-z, A-Z, 0-9, space, and the following characters: `_ . : / = + -` and `@`
- Schlüssel und Werte unterscheiden zwischen Groß- und Kleinschreibung.
- Verwenden Sie nicht `aws :` als Präfix für den Schlüssel. Dieses Präfix ist für AWS reserviert.

Verwaltung von Tags in AWS Elemental MediaTailor

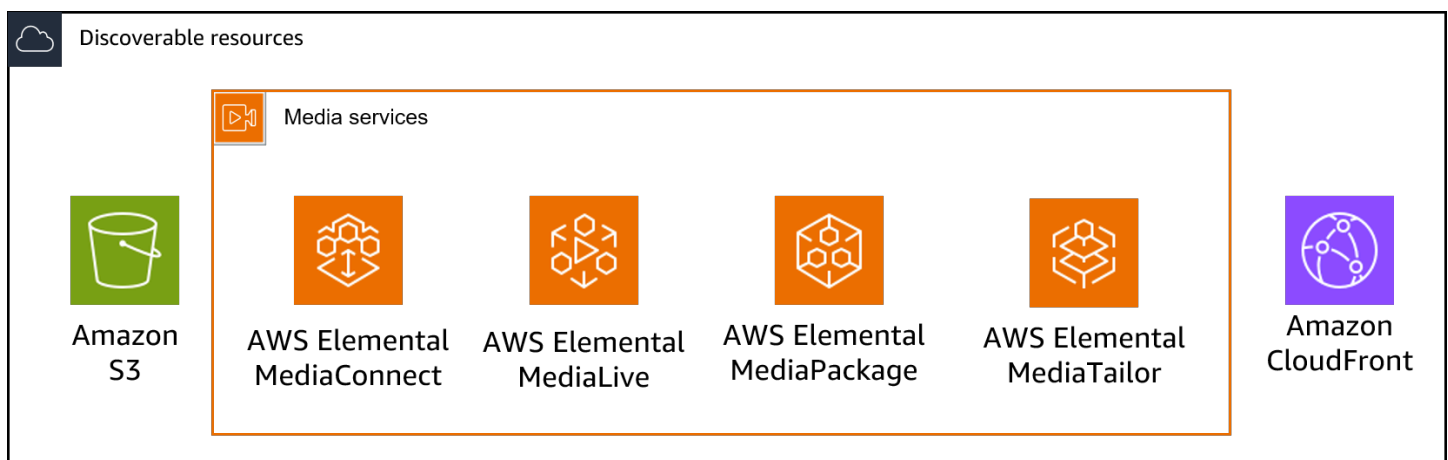
Sie legen Tags als Eigenschaften für eine Ressource fest. Sie können Tags über die AWS Elemental MediaTailor API oder die AWS Command Line Interface (AWS CLI) hinzufügen, bearbeiten und löschen. Weitere Informationen finden Sie in der [AWS Elemental MediaTailor API-Referenz](#).

Überwachung von AWS Mediendiensten mit Workflow-Monitor

Der Workflow-Monitor ist ein Tool zur Erkennung, Visualisierung und Überwachung von AWS Medien-Workflows. Der Workflow-Monitor ist in der AWS Konsole und in der API verfügbar. Sie können den Workflow-Monitor verwenden, um visuelle Zuordnungen der Ressourcen Ihres Workflows, sogenannte Signalzuordnungen, zu ermitteln und zu erstellen. Sie können CloudWatch Amazon-Alarm- und EventBridge Amazon-Regelvorlagen erstellen und verwalten, um die zugewiesenen Ressourcen zu überwachen. Die von Ihnen erstellten Überwachungsvorlagen werden in bereitstellbare AWS CloudFormation Vorlagen umgewandelt, um Wiederholbarkeit zu gewährleisten. AWS empfohlene Alarmvorlagen bieten vordefinierte Best-Practice-Überwachung.

Entdecken

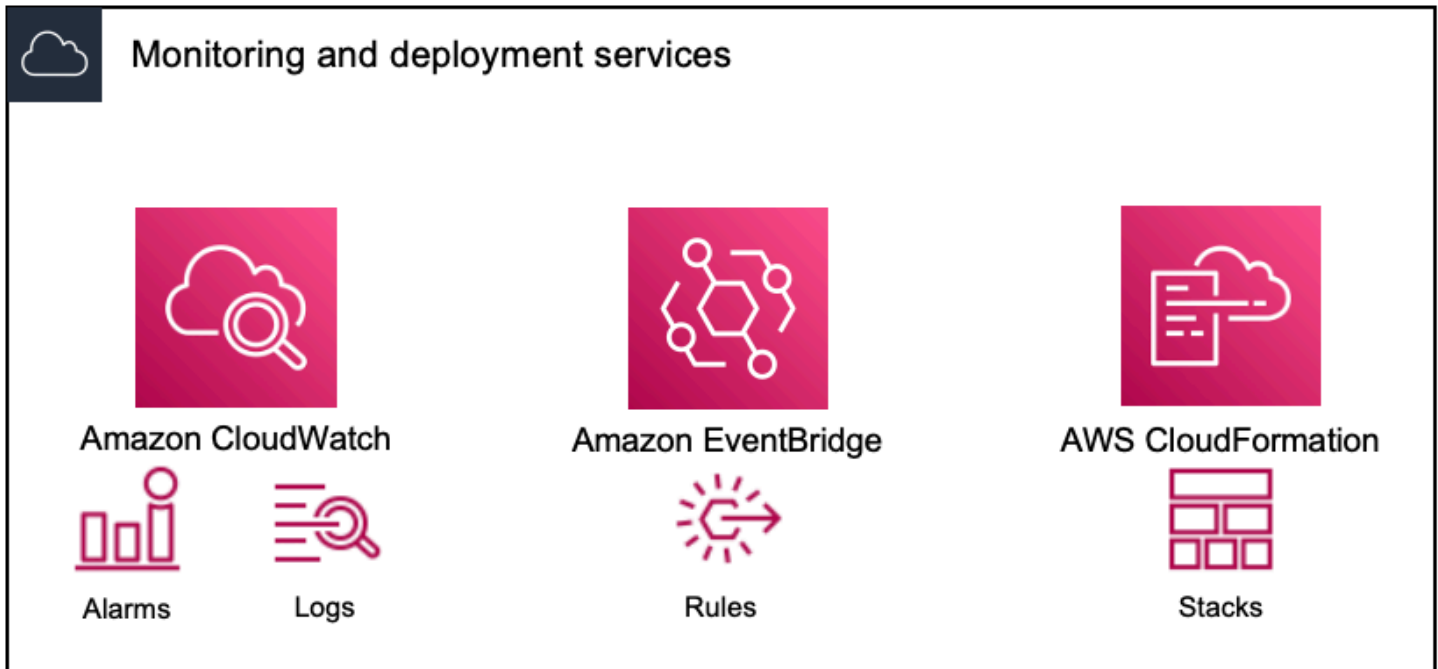
Nutzen Sie Signalkarten, um automatisch miteinander verbundene AWS Ressourcen zu erkennen, die mit Ihrem Medien-Workflow verknüpft sind. Die Erkennung kann bei jeder unterstützten Servicereource beginnen und erstellt eine end-to-end Zuordnung des Workflows. Signalkarten können als eigenständige Visualisierungstools verwendet oder mit Überwachungsvorlagen erweitert werden.



Monitor

Sie können benutzerdefinierte CloudWatch Alarm- und EventBridge Regelvorlagen erstellen, um den Zustand und den Status Ihrer Medien-Workflows zu überwachen. Es stehen bewährte

Alarmvorlagen für den Import in Ihre Workflow-Monitor-Umgebung zur Verfügung. Sie können die Best-Practice-Alarmvorlagen unverändert verwenden oder sie so bearbeiten, dass sie besser zu Ihrem Arbeitsablauf passen. Alle Vorlagen, die Sie erstellen, werden in AWS CloudFormation Vorlagen für eine wiederholbare Bereitstellung umgewandelt.



Note

Für die Verwendung des Workflow-Monitors fallen keine direkten Kosten an. Es fallen jedoch Kosten im Zusammenhang mit den Ressourcen an, die zur Überwachung Ihres Workflows erstellt und verwendet werden.

Wenn die Überwachung bereitgestellt wird, werden Amazon CloudWatch - und EventBridge Amazon-Ressourcen erstellt. Wenn Sie die AWS Managementkonsole verwenden, werden Sie vor der Bereitstellung der Überwachung auf einer Signalkarte darüber informiert, wie viele Ressourcen erstellt werden. Weitere Informationen zur Preisgestaltung finden Sie unter: [CloudWatchPreise](#) und [EventBridge Preisgestaltung](#).

Der Workflow-Monitor verwendet AWS CloudFormation Vorlagen für die Bereitstellung der CloudWatch EventBridge Ressourcen. Diese Vorlagen werden in einem Amazon Simple Storage Service-Bucket der Standardklasse gespeichert, der während des Bereitstellungsprozesses von Workflow-Monitor in Ihrem Namen erstellt wird und für den Objektspeicher und Rückruf Gebühren anfallen. Weitere Informationen zur Preisgestaltung finden Sie unter: [Amazon S3 S3-Preise](#).

In der Workflow-Monitor-Signalmap für AWS Elemental MediaPackage Kanäle generierte Vorschauen werden vom MediaPackage Origin-Endpunkt aus bereitgestellt und es fallen Gebühren für ausgehende Datenübertragungen an. [Die Preise findest du unter: MediaPackage Preise.](#)

Komponenten des Workflow-Monitors

Der Workflow-Monitor besteht aus vier Hauptkomponenten:

- CloudWatch Alarmvorlagen — Definieren Sie die Bedingungen, anhand derer Sie die Überwachung durchführen möchten CloudWatch. Sie können Ihre eigenen Alarmvorlagen erstellen oder vordefinierte Vorlagen importieren, die von erstellt wurden AWS. Weitere Informationen finden Sie unter: [CloudWatch Alarmgruppen und Vorlagen zur Überwachung Ihres AWS Medienworkflows](#)
- EventBridge Regelvorlagen — Definieren Sie, wie Benachrichtigungen EventBridge gesendet werden, wenn ein Alarm ausgelöst wird. Weitere Informationen finden Sie unter: [EventBridge Regelgruppen und Vorlagen zur Überwachung Ihres AWS Medien-Workflows](#)
- Signalzuordnungen — Verwenden Sie einen automatisierten Prozess, um anhand vorhandener AWS Ressourcen AWS elementare Workflow-Übersichten zu erstellen. Die Signalzuordnungen können verwendet werden, um Ressourcen in Ihrem Workflow zu erkennen und diese Ressourcen zu überwachen. Weitere Informationen finden Sie unter: [Signalkarten des Workflow-Monitors](#)
- Überblick — Auf der Übersichtsseite können Sie den Status mehrerer Signalkarten direkt von einem Standort aus überwachen. Überprüfen Sie Metriken, Protokolle und Alarmer für Ihre Workflows. Weitere Informationen finden Sie unter: [Überblick über den Workflow-Monitor](#)

Unterstützte Services

Der Workflow-Monitor unterstützt die automatische Erkennung und Signalzuweisung von Ressourcen, die mit den folgenden Diensten verknüpft sind:

- AWS Elemental MediaConnect
- AWS Elemental MediaLive
- AWS Elemental MediaPackage
- AWS Elemental MediaTailor
- Amazon S3
- Amazon CloudFront

Themen

- [Workflow-Monitor zur Überwachung von AWS Mediendiensten konfigurieren](#)
- [Verwenden Sie den Workflow-Monitor](#)

Workflow-Monitor zur Überwachung von AWS Mediendiensten konfigurieren

Um den Workflow-Monitor zum ersten Mal einzurichten, erstellen Sie die Alarm- und Ereignisvorlagen und entdecken Signalkarten, die zur Überwachung Ihrer Medien-Workflows verwendet werden. Die folgende Anleitung enthält die erforderlichen Schritte, um IAM-Rollen auf Administrator- und Operator-Ebene einzurichten, Workflow-Monitor-Ressourcen zu erstellen und die Überwachung für Ihre Workflows bereitzustellen.

Themen

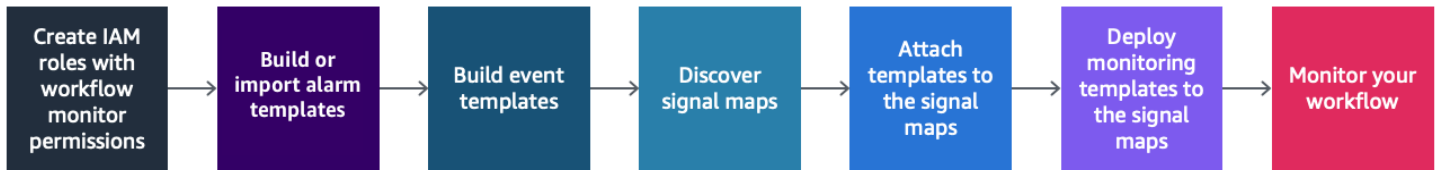
- [Erste Schritte mit dem Workflow-Monitor](#)
- [Workflow-Monitor-Gruppen und -Vorlagen](#)
- [Signalkarten des Workflow-Monitors](#)
- [Workflow-Monitor-Kontingente](#)

Erste Schritte mit dem Workflow-Monitor

Die folgenden Schritte bieten einen grundlegenden Überblick über die erstmalige Verwendung von Workflow Monitor.

1. Richten Sie Workflow-Monitor-IAM-Berechtigungen für Rollen auf Administrator- und Operatorebene ein: [IAM-Richtlinien zur Workflow-Überwachung](#)
2. Erstellen Sie Alarmvorlagen oder importieren Sie vordefinierte Vorlagen, die erstellt wurden von AWS: [CloudWatch-Alarme](#)
3. Erstellen Sie Benachrichtigungsereignisse, die zugestellt werden von EventBridge: [EventBridge Regeln](#)
4. Entdecken Sie Signalkarten mithilfe Ihrer vorhandenen AWS Elementarressourcen: [Signalkarten](#)
5. Hängen Sie die Alarmvorlagen und Benachrichtigungsregeln an Ihre Signalkarte an: [Vorlagen anhängen](#)
6. Stellen Sie die Vorlagen bereit, um mit der Überwachung der Signalkarte zu beginnen: [Bereitstellung von Vorlagen für die Überwachung](#)

7. Überwachen und überprüfen Sie Ihre Workflow-Monitor-Ressourcen mithilfe des Übersichtsbereichs der AWS Konsole: [Übersicht](#)



Workflow-Monitor (IAM-Richtlinien)

Der Workflow-Monitor interagiert mit mehreren AWS Diensten, um Signalkarten, Builds CloudWatch und EventBridge Ressourcen sowie Vorlagen zu erstellen. AWS CloudFormation Da der Workflow-Monitor mit einer Vielzahl von Diensten interagiert, müssen für diese Dienste spezifische AWS Identity and Access Management (IAM-) Richtlinien zugewiesen werden. In den folgenden Beispielen werden die erforderlichen IAM-Richtlinien sowohl für Administrator- als auch für Operator-IAM-Rollen beschrieben.

IAM-Richtlinie für Administratoren

Die folgende Beispielrichtlinie bezieht sich auf eine IAM-Richtlinie für den Workflow-Monitor auf Administratorebene. Diese Rolle ermöglicht die Erstellung und Verwaltung von Workflow-Monitor-Ressourcen und der unterstützten Servicere Ressourcen, die mit Workflow-Monitor interagieren.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:List*",
        "cloudwatch:Describe*",
        "cloudwatch:Get*",
        "cloudwatch:PutAnomalyDetector",
        "cloudwatch:PutMetricData",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch:PutCompositeAlarm",
        "cloudwatch:PutDashboard",
        "cloudwatch:DeleteAlarms",
        "cloudwatch:DeleteAnomalyDetector",
      ]
    }
  ]
}

```

```

        "cloudwatch:DeleteDashboards",
        "cloudwatch:TagResource",
        "cloudwatch:UntagResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "cloudformation:List*",
        "cloudformation:Describe*",
        "cloudformation:CreateStack",
        "cloudformation:UpdateStack",
        "cloudformation>DeleteStack",
        "cloudformation:TagResource",
        "cloudformation:UntagResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "cloudfront:List*",
        "cloudfront:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "events:List*",
        "events:Describe*",
        "events:CreateEventBus",
        "events:PutRule",
        "events:PutTargets",
        "events:EnableRule",
        "events:DisableRule",

```

```

        "events:DeleteRule",
        "events:RemoveTargets",
        "events:TagResource",
        "events:UntagResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "logs:Describe*",
        "logs:Get*",
        "logs:TagLogGroup",
        "logs:TagResource",
        "logs:UntagLogGroup",
        "logs:UntagResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediaconnect:List*",
        "mediaconnect:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "medialive:*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackage:List*",
        "mediapackage:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",

```

```

    "Action": [
      "mediapackagev2:List*",
      "mediapackagev2:Get*"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "mediapackage-vod:List*",
      "mediapackage-vod:Describe*"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "mediatailor:List*",
      "mediatailor:Describe*",
      "mediatailor:Get*"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "resource-groups:ListGroup",
      "resource-groups:GetGroup",
      "resource-groups:GetTags",
      "resource-groups:GetGroupQuery",
      "resource-groups:GetGroupConfiguration",
      "resource-groups:CreateGroup",
      "resource-groups:UngroupResources",
      "resource-groups:GroupResources",
      "resource-groups>DeleteGroup",
      "resource-groups:UpdateGroupQuery",
      "resource-groups:UpdateGroup",
      "resource-groups:Tag",
      "resource-groups:Untag"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",

```

```

    "Action": [
      "s3:*"
    ],
    "Resource": "arn:aws:s3:::workflow-monitor-templates*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "sns:TagResource",
      "sns:UntagResource"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "tag:Get*",
      "tag:Describe*",
      "tag:TagResources",
      "tag:UntagResources"
    ],
    "Resource": "*"
  }
]
}

```

IAM-Richtlinie für Betreiber

Die folgende Beispielrichtlinie bezieht sich auf eine IAM-Richtlinie für den Workflow-Monitor auf Operatorebene. Diese Rolle ermöglicht eingeschränkten und schreibgeschützten Zugriff auf die Workflow-Monitor-Ressourcen und die unterstützten Servicere Ressourcen, die mit dem Workflow-Monitor interagieren.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [

```

```

        "cloudwatch:List*",
        "cloudwatch:Describe*",
        "cloudwatch:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "cloudformation:List*",
        "cloudformation:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "cloudfront:List*",
        "cloudfront:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "events:List*",
        "events:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "logs:Describe*",
        "logs:Get*"
    ],
    "Resource": "*"
}

```

```
    },
    {
      "Effect": "Allow",
      "Action": [
        "mediaconnect:List*",
        "mediaconnect:Describe*"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "medialive:List*",
        "medialive:Get*",
        "medialive:Describe*"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "mediapackage:List*",
        "mediapackage:Describe*"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "mediapackagev2:List*",
        "mediapackagev2:Get*"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "mediapackage-vod:List*",
        "mediapackage-vod:Describe*"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
```



```

    "Action": [
      "mediatailor:List*",
      "mediatailor:Describe*",
      "mediatailor:Get*"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "s3:Get*",
      "s3:List*"
    ],
    "Resource": "arn:aws:s3:::workflow-monitor-templates*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "tag:Get*",
      "tag:Describe*"
    ],
    "Resource": "*"
  }
]
}

```

Workflow-Monitor-Gruppen und -Vorlagen

Bevor Sie die Workflow-Überwachung in einer Signalmap einsetzen können, müssen Sie die Gruppen und Vorlagen für CloudWatch Alarme und EventBridge Benachrichtigungen erstellen. Die CloudWatch Vorlagen definieren, welche Szenarien und Schwellenwerte verwendet werden, um die Alarme auszulösen. Die EventBridge Vorlagen bestimmen, wie Ihnen diese Alarme gemeldet werden.

Wenn Sie nur Zuordnungen Ihrer verbundenen Ressourcen wünschen und nicht die Funktionen der Überwachungsvorlagen von Workflow-Monitor verwenden möchten, können Sie Signalzuordnungen auch ohne CloudWatch Vorlagen verwenden. EventBridge Weitere Informationen zur Verwendung von Signalkarten finden Sie unter: [Signalkarten](#)

Themen

- [CloudWatch Alarmgruppen und Vorlagen zur Überwachung Ihres AWS Medienworkflows](#)

- [EventBridge Regelgruppen und Vorlagen zur Überwachung Ihres AWS Medien-Workflows](#)

CloudWatch Alarmgruppen und Vorlagen zur Überwachung Ihres AWS Medienworkflows

Mit Workflow-Monitor-Alarmen können Sie vorhandene CloudWatch Messwerte als Grundlage für Alarme für Ihre Signalkarten verwenden. Sie können eine Alarmvorlagengruppe erstellen, um die Alarmtypen zu sortieren und zu klassifizieren, die für Ihren Arbeitsablauf wichtig sind. Innerhalb jeder Alarmvorlagengruppe erstellen Sie Alarmvorlagen mit bestimmten CloudWatch Metriken und Parametern, die Sie überwachen möchten. Sie können Ihre eigenen Alarmvorlagen erstellen oder empfohlene Alarmvorlagen importieren, die von erstellt wurden AWS. Nachdem Sie eine Alarmvorlagengruppe und Alarmvorlagen innerhalb dieser Gruppe erstellt haben, können Sie eine oder mehrere dieser Alarmvorlagengruppen an eine Signalmap anhängen.

Sie müssen zuerst eine Alarmvorlagengruppe erstellen. Nachdem Sie eine Alarmvorlagengruppe erstellt haben, können Sie Ihre eigenen Vorlagen erstellen oder empfohlene Vorlagen verwenden, die von erstellt wurden AWS. Wenn Sie Ihre eigenen Alarmvorlagen erstellen möchten, fahren Sie auf dieser Seite fort. Weitere Informationen zum Importieren von empfohlenen Vorlagen finden Sie unter: [Empfohlene Vorlagen](#)

In diesem Abschnitt wird die Erstellung von CloudWatch Alarmen mithilfe des Workflow-Monitors behandelt. Weitere Informationen darüber, wie der CloudWatch Service mit Alarmen umgeht, und Einzelheiten zu den Alarmkomponenten finden Sie unter: [Verwenden von CloudWatch Alarmen](#) im CloudWatch Amazon-Benutzerhandbuch

Alarm-Vorlagengruppen erstellen

Mithilfe von Alarmvorlagengruppen können Sie die Alarmtypen sortieren und klassifizieren, die für Ihren Arbeitsablauf wichtig sind.

Um eine Alarmvorlagengruppe zu erstellen

1. Wählen Sie im Navigationsbereich der Workflow-Monitor-Konsole die Option CloudWatchAlarmvorlagen aus.
2. Wählen Sie Alarmvorlagengruppe erstellen aus.
3. Geben Sie der Alarmvorlagengruppe einen eindeutigen Gruppennamen und optional eine Beschreibung.
4. Wählen Sie Erstellen aus. Sie werden zur Detailseite der neu erstellten Alarmvorlagengruppe weitergeleitet.

Alarmvorlagen erstellen

Sie können Alarmvorlagen mit den CloudWatch Metriken und Parametern erstellen, die Sie überwachen möchten.

Um eine Alarmvorlage zu erstellen

1. Wählen Sie auf der Detailseite der Alarmvorlagengruppe die Option Alarmvorlage erstellen aus.
2. Geben Sie der Alarmvorlage einen eindeutigen Vorlagennamen und optional eine Beschreibung.
3. Gehen Sie im Abschnitt Metrik auswählen wie folgt vor:
 1. Wählen Sie einen Zielressourcentyp aus. Der Zielressourcentyp ist eine Ressource für den jeweiligen Service, z. B. ein Kanal für MediaLive und MediaPackage oder ein Flow für MediaConnect.
 2. Wählen Sie einen Metriknamen aus. Dies ist die CloudWatch Metrik, die als Grundlage für den Alarm dient. Die Liste der Metriken ändert sich je nach ausgewähltem Zielressourcentyp.
4. Gehen Sie im Bereich Alarmeinstellungen wie folgt vor:

Note

Weitere Informationen darüber, wie der CloudWatch Service mit Alarmen umgeht, und Einzelheiten zu den Alarmkomponenten finden Sie unter: [Verwenden von CloudWatch Alarmen](#) im CloudWatch Amazon-Benutzerhandbuch

1. Wählen Sie die Statistik aus. Dies ist ein Wert, z. B. eine Summe oder ein Durchschnitt, der zur Überwachung der Metrik verwendet wird.
2. Wählen Sie den Vergleichsoperator aus. Dieses Feld verweist auf den Schwellenwert, den Sie im nächsten Schritt festgelegt haben.
3. Legen Sie einen Schwellenwert fest. Dies ist ein numerischer Wert, anhand dessen der Vergleichsoperator bestimmt, ob der Status größer als, kleiner oder gleich dem Status ist.
4. Legen Sie einen Zeitraum fest. Dies ist ein Zeitwert in Sekunden. Der Zeitraum ist der Zeitraum, in dem Statistik, Vergleichsoperator und Schwellenwert interagieren, um zu bestimmen, ob der Alarm ausgelöst wird.
5. Legen Sie die Datenpunkte fest. Dieser Wert bestimmt, wie viele Datenpunkte benötigt werden, um den Alarm auszulösen.

6. Wählen Sie aus, wie fehlende Daten behandelt werden sollen. Diese Auswahl bestimmt, wie dieser Alarm auf fehlende Daten reagiert.
5. Wählen Sie Erstellen aus, um den Vorgang abzuschließen.

Ein Beispiel für eine fertige Alarmvorlage könnte die folgenden Parameter haben: Ein MediaConnect Flow-Zielressourcentyp wird auf den Metriknamen für Verbindungsabbrüche überwacht. Der Statistikwert ist auf Summe mit einem Vergleichsoperator von „größer als oder gleich“ und einem Schwellenwert von 10 festgelegt. Der Zeitraum ist auf 60 Sekunden festgelegt und erfordert nur 1 von 1 Datenpunkten. Fehlende Daten behandeln ist auf „Ignorieren“ gesetzt.

Das Ergebnis dieser Einstellungen ist: Der Workflow-Monitor überwacht, ob es im Flow zu Verbindungsabbrüchen kommt. Wenn innerhalb von 60 Sekunden 10 oder mehr Verbindungsabbrüche auftreten, wird der Alarm ausgelöst. 10 oder mehr Verbindungsabbrüche innerhalb von 60 Sekunden müssen nur einmal geschehen, damit der Alarm ausgelöst wird.

Empfohlene Alarmvorlagen für die Überwachung Ihres Medienworkflows AWS

Bei den empfohlenen Vorlagen von Workflow Monitor handelt es sich um eine kuratierte Auswahl von AWS Elemental-Servicemetriken mit vordefinierten Alarmeinstellungen, die für die Metrik geeignet sind. Wenn Sie keine benutzerdefinierten Alarmvorlagen erstellen möchten, bieten Ihnen die empfohlenen Vorlagen Best-Practice-Überwachungsvorlagen, die von erstellt wurden. AWS

Der Workflow-Monitor enthält empfohlene Vorlagengruppen für jeden unterstützten Dienst. Diese Gruppen sind so konzipiert, dass sie die Überwachung nach bewährten Methoden auf bestimmte Arten von Workflows anwenden. Jede Vorlagengruppe enthält eine kuratierte Auswahl von Alarmen, die anhand dienstspezifischer Metriken konfiguriert wurden. Beispielsweise hat eine empfohlene Vorlagengruppe für einen MediaLive Multiplex-Workflow einen anderen Satz vorkonfigurierter Metriken als ein CDI-Workflow. MediaConnect

Um empfohlene Alarmvorlagen zu verwenden

1. Folgen Sie den Schritten, um [eine Alarmvorlagengruppe zu erstellen](#), oder wählen Sie eine bestehende aus.
2. Wählen Sie im Abschnitt Alarmvorlagen die Option Importieren aus. Sie müssen die AWS empfohlenen Vorlagen in Ihre Vorlagengruppe importieren.
3. Verwenden Sie das Drop-down-Menü für CloudWatch Alarmvorlagengruppen, um eine AWS empfohlene Gruppe auszuwählen. Diese Gruppen enthalten kuratierte Alarme für bestimmte Dienste.

4. Wählen Sie mithilfe der Kontrollkästchen die Vorlagen aus, die importiert werden sollen. Jede Vorlage listet ihre Metriken und vorkonfigurierten Überwachungswerte auf und bietet eine Beschreibung der Metrik. Wenn Sie mit der Auswahl der Vorlagen fertig sind, klicken Sie auf die Schaltfläche Hinzufügen.
5. Die ausgewählten Vorlagen werden in den Bereich „Zu importierende Alarmvorlage (n)“ verschoben. Überprüfen Sie Ihre Auswahl und wählen Sie Importieren aus.
6. Nach Abschluss des Imports werden die ausgewählten Vorlagen der Vorlagengruppe hinzugefügt. Wenn Sie weitere Vorlagen hinzufügen möchten, wiederholen Sie den Importvorgang.
7. Importierte Vorlagen können nach dem Import angepasst werden. Die Alarmeinstellungen können an Ihre Alarmanforderungen angepasst werden.

EventBridge Regelgruppen und Vorlagen zur Überwachung Ihres AWS Medien-Workflows

CloudWatch verwendet EventBridge Amazon-Regeln zum Senden von Benachrichtigungen. Sie beginnen mit der Erstellung einer Event-Vorlagengruppe. In dieser Vorlagengruppe für Ereignisse erstellen Sie Ereignisvorlagen, die festlegen, unter welchen Bedingungen eine Benachrichtigung ausgelöst wird und wer benachrichtigt wird.

In diesem Abschnitt wird die Erstellung von EventBridge Regeln mithilfe des Workflow-Monitors behandelt. Weitere Informationen darüber, wie der EventBridge Service Regeln verwendet, finden Sie unter: [EventBridge Regeln](#) im EventBridge Amazon-Benutzerhandbuch

Vorlagengruppen für Veranstaltungen erstellen

Mithilfe von Vorlagengruppen für Ereignisse können Sie Ereignisse anhand Ihres Anwendungsfalls sortieren und klassifizieren.

Um eine Vorlagengruppe für Ereignisse zu erstellen

1. Wählen Sie im Navigationsbereich der Workflow-Monitor-Konsole EventBridge Regelvorlagen aus.
2. Wählen Sie Ereignisvorlagengruppe erstellen aus.
3. Geben Sie der Alarmvorlagengruppe einen eindeutigen Gruppennamen und optional eine Beschreibung.
4. Wählen Sie Erstellen aus. Sie werden zur Detailseite der neu erstellten Alarmvorlagengruppe weitergeleitet.

Event-Vorlagen erstellen

Sie können Benachrichtigungen auf der Grundlage von Ereignisvorlagen senden, die Sie erstellt haben.

Um eine Event-Vorlage zu erstellen

1. Wählen Sie auf der Detailseite der Veranstaltungsvorlagengruppe die Option Eventvorlage erstellen aus.
2. Geben Sie der Veranstaltungsvorlage einen eindeutigen Vorlagennamen und optional eine Beschreibung.
3. Gehen Sie im Abschnitt Regeleinstellungen wie folgt vor:
 1. Wählen Sie einen Ereignistyp aus. Bei der Auswahl eines Ereignistyps können Sie zwischen mehreren Ereignissen wählen, die von einem Alarm erstellt wurden, AWS oder die Option Signal map active auswählen, um einen Alarm zu verwenden, der mit einer Alarmvorlage erstellt wurde.
 2. Wählen Sie einen Target-Service aus. Dies bestimmt, wie Sie über dieses Ereignis informiert werden möchten. Sie können Amazon Simple Notification Service oder CloudWatch Protokolle auswählen.
 3. Nachdem Sie einen Zieldienst ausgewählt haben, wählen Sie ein Ziel aus. Dies ist ein Amazon SNS SNS-Thema oder eine CloudWatch Protokollgruppe, abhängig von Ihrer Zielservice-Auswahl.
4. Wählen Sie Erstellen aus, um den Vorgang abzuschließen.

Signalkarten des Workflow-Monitors

Signalkarten sind visuelle Zuordnungen von AWS Ressourcen in Ihrem Medienworkflow. Sie können den Workflow-Monitor verwenden, um die Signalzuordnungssuche für jeden der unterstützten Ressourcentypen zu starten. Während des Erkennungsprozesses ordnet der Workflow-Monitor automatisch und rekursiv alle verbundenen AWS Ressourcen zu. Nachdem die Signalmap erstellt wurde, können Sie die Workflow-Monitor-Konsole verwenden, um beispielsweise Überwachungsvorlagen bereitzustellen, Metriken anzuzeigen und Details der zugewiesenen Ressourcen anzuzeigen.

Themen

- [Signalkarten für AWS Medien-Workflows erstellen](#)

- [Signalkarten von AWS Medien-Workflows anzeigen](#)
- [Hinzufügen von Alarm- und Ereignisvorlagen zur Signalübersicht Ihres AWS Medienworkflows](#)
- [Bereitstellung von Vorlagen für die Signallampe Ihres AWS Medienworkflows](#)
- [Aktualisierung der Signalkarte Ihres AWS Medien-Workflows](#)
- [Löschen der Signallampe Ihres AWS Medien-Workflows](#)

Signalkarten für AWS Medien-Workflows erstellen

Sie können Workflow-Monitor-Signalzuordnungen verwenden, um eine visuelle Zuordnung aller verbundenen AWS Ressourcen in Ihrem Medien-Workflow zu erstellen.

Um eine Signalmap zu erstellen

1. Wählen Sie im Navigationsbereich der Workflow-Monitor-Konsole die Option Signalzuordnungen aus.
2. Wählen Sie Signalkarte erstellen aus.
3. Geben Sie der Signalmap einen Namen und eine Beschreibung.
4. Im Abschnitt Neue Signalkarte entdecken werden die Ressourcen im aktuellen Konto und in der ausgewählten Region angezeigt. Wählen Sie eine Ressource aus, um mit der Erkennung der Signalkarte zu beginnen. Die ausgewählte Ressource wird der Ausgangspunkt für die Suche sein.
5. Wählen Sie Erstellen aus. Warten Sie einen Moment, bis der Entdeckungsprozess abgeschlossen ist. Nach Abschluss des Vorgangs wird Ihnen die neue Signalkarte angezeigt.

Note

In der Workflow-Monitor-Signalmap für AWS Elemental MediaPackage Kanäle generierte Vorschauen werden vom MediaPackage Origin-Endpunkt aus bereitgestellt und es fallen Gebühren für ausgehende Datenübertragungen an. [Die Preise findest du unter: MediaPackage Preise.](#)

Signalkarten von AWS Medien-Workflows anzeigen

Mithilfe der Signalkarten des Workflow-Monitors können Sie sich eine visuelle Zuordnung aller verbundenen AWS Ressourcen in Ihrem Medien-Workflow anzeigen lassen.

Signalkartenansichten

Nachdem Sie eine Signalkarte ausgewählt haben, stehen Ihnen zwei Ansichten zur Verfügung, mit denen Sie die Signalkarte überwachen oder konfigurieren können. Signal-Map überwachen und Signal-Map konfigurieren ist eine kontextsensitive Schaltfläche, die sich oben rechts im Bereich der Signal-Map-Konsole befindet.

Wenn Sie die Signalkarte im Abschnitt Signalkarten des Navigationsfensters auswählen, wird Ihre Signalkarte in der Konfigurationsansicht angezeigt. In der Konfigurationsansicht können Sie Änderungen an den Vorlagengruppen vornehmen, die an diese Signalkarte angehängt sind, die angehängten Vorlagen bereitstellen und die grundlegenden Details und Tags der Signalkarte anzeigen.

Wenn Sie die Signalkarte im Abschnitt Übersicht des Navigationsfensters auswählen, wird Ihre Signalkarte in der Überwachungsansicht angezeigt. In der Überwachungsansicht werden die CloudWatch Alarmer, EventBridge Regeln, Warnungen, Protokolle und Messwerte für diese Signalkarte angezeigt.

Die Ansicht kann jederzeit geändert werden, indem Sie oben rechts auf die Schaltfläche Signalmap überwachen/konfigurieren klicken. Für die Konfigurationsansicht sind IAM-Berechtigungen auf Administratorebene erforderlich. Die erforderlichen IAM-Berechtigungen können hier eingesehen werden: [IAM-Richtlinien zur Workflow-Überwachung](#)

In der Signalkarte navigieren

Eine Signalmap enthält Knoten für jede unterstützte AWS-Ressource, die vom Workflow-Monitor erkannt wurde. Bestimmte Ressourcen, wie MediaLive Kanäle und MediaPackage Endpunkte, können Vorschaubilder des Inhalts anzeigen, sofern Vorschaubilder verfügbar sind.

Wenn Sie einen Ressourcenknoten auswählen und im Dropdownmenü Aktionen die Option Ausgewählte Ressourcendetails anzeigen auswählen, gelangen Sie zur Detailseite des zugehörigen Dienstes. Wenn Sie beispielsweise einen MediaLive Kanal auswählen und Ausgewählte Ressourcendetails anzeigen auswählen, wird die Detailseite der MediaLive Konsole für diesen Kanal geöffnet.

Wenn Sie einen Ressourcenknoten auswählen, wird die Liste der aktiven Alarmer nur auf diesen Knoten gefiltert. Wenn Sie im aktiven Alarm den Ziel-ARN der Ressource auswählen, werden Sie zur Detailseite des zugehörigen Dienstes weitergeleitet, auf der die ausgewählte Ressource geöffnet ist.

Hinzufügen von Alarm- und Ereignisvorlagen zur Signalübersicht Ihres AWS Medienworkflows

Nachdem Sie Alarm- und Ereignisvorlagen erstellt haben, müssen Sie diese an eine Signalkarte anhängen. Jede der von Ihnen erstellten Alarm- und Ereignisvorlagen kann an alle erkannten Signalkarten angehängt werden.

Um Alarm- und Ereignisvorlagen an Ihre Signalkarte anzuhängen

1. Wählen Sie im Navigationsbereich der Workflow-Monitor-Konsole die Option Signalzuordnungen und dann die Signalkarte aus, mit der Sie arbeiten möchten.
2. Wählen Sie oben rechts auf der Signalzuordnungsseite auf der Registerkarte CloudWatchAlarmvorlagengruppen die Option Alarmvorlagengruppen anhängen CloudWatch aus.
 1. Wählen Sie in dem neuen Bereich, der geöffnet wird, alle Alarmvorlagengruppen aus, die Sie auf diese Signalzuordnung anwenden möchten, und wählen Sie dann Hinzufügen aus. Dadurch werden die ausgewählten Alarmvorlagengruppen in den Abschnitt Angehängte CloudWatch Alarmvorlagengruppen verschoben.
 2. Wenn Sie Speichern auswählen, werden Ihre Änderungen gespeichert und Sie kehren zur Signalzuordnungsseite zurück.
3. Wählen Sie rechts auf der Signalzuordnungsseite die Registerkarte EventBridge Regelvorlagengruppen und dann EventBridgeRegelvorlagengruppen anhängen aus.
 1. Wählen Sie im neuen Bereich, der geöffnet wird, alle Event-Vorlagengruppen aus, die Sie auf diese Signalmap anwenden möchten, und wählen Sie dann Hinzufügen aus. Dadurch werden die ausgewählten Regelvorlagengruppen in den Abschnitt Angehängte EventBridge Regelvorlagengruppen verschoben.
 2. Wenn Sie Speichern auswählen, werden Ihre Änderungen gespeichert und Sie kehren zur Signalzuordnungsseite zurück.
4. Sie haben der Signalkarte CloudWatch Alarm- und EventBridge Regelvorlagen zugewiesen, aber die Überwachung ist noch nicht implementiert. Im nächsten Abschnitt wird der Einsatz der Überwachungsressourcen behandelt.

Bereitstellung von Vorlagen für die Signallampe Ihres AWS Medienworkflows

Nachdem Sie die Alarm- und Ereignisvorlagen an Ihre Signalkarte angehängt haben, müssen Sie die Überwachung bereitstellen. Bis die Bereitstellung abgeschlossen ist, ist die Überwachung Ihrer Signalkarte nicht aktiv.

Der Workflow-Monitor löst nur Alarme aus, die für die ausgewählte Signallampe relevant sind. Beispielsweise kann die angehängte Alarmvorlagengruppe Alarme für mehrere Dienste enthalten, z. B. MediaLive MediaPackage, und MediaConnect. Wenn die ausgewählte Signalzuordnung nur MediaLive Ressourcen enthält, werden keine MediaPackage MediaConnect Alarme ausgelöst.

Um die Überwachungsvorlagen bereitzustellen

1. Nachdem Sie Ihrer Signalübersicht Gruppen mit Alarm- und Ereignisvorlagen hinzugefügt und Ihre Änderungen gespeichert haben, wählen Sie im Dropdownmenü Aktionen die Option Monitor bereitstellen aus.
2. Sie werden aufgefordert, die Bereitstellung zu bestätigen, und es werden die Anzahl CloudWatch und die EventBridge Ressourcen angezeigt, die erstellt werden sollen. Wenn Sie fortfahren möchten, wählen Sie Bereitstellen aus.

Note

Für die Nutzung des Workflow-Monitors fallen keine direkten Kosten an. Es fallen jedoch Kosten im Zusammenhang mit den Ressourcen an, die zur Überwachung Ihres Workflows erstellt und verwendet werden.

Wenn die Überwachung bereitgestellt wird, werden Amazon CloudWatch - und EventBridge Amazon-Ressourcen erstellt. Wenn Sie die AWS Managementkonsole verwenden, werden Sie vor der Bereitstellung der Überwachung auf einer Signalkarte darüber informiert, wie viele Ressourcen erstellt werden. Weitere Informationen zur Preisgestaltung finden Sie unter: [CloudWatchPreise](#) und [EventBridge Preisgestaltung](#). Der Workflow-Monitor verwendet AWS CloudFormation Vorlagen für die Bereitstellung der CloudWatch EventBridge Ressourcen. Diese Vorlagen werden in einem Amazon Simple Storage Service-Bucket der Standardklasse gespeichert, der während des Bereitstellungsprozesses von Workflow-Monitor in Ihrem Namen erstellt wird und für den Objektspeicher und Rückruf Gebühren anfallen. Weitere Informationen zur Preisgestaltung finden Sie unter: [Amazon S3 S3-Preise](#).

3. Der Status der Bereitstellung wird neben dem Namen der Signalkarte angezeigt. Der Bereitstellungsstatus ist auch im Bereich Stacks der AWS CloudFormation Konsole sichtbar. Nach einigen Augenblicken der Erstellung und Bereitstellung der Ressourcen beginnt die Überwachung Ihrer Signalkarte.

Aktualisierung der Signalkarte Ihres AWS Medien-Workflows

Wenn an Ihrem Arbeitsablauf eine Änderung vorgenommen wird, müssen Sie möglicherweise die Signalkarte erneut ermitteln und die Überwachungsressourcen erneut einsetzen. Der Workflow-Monitor ist ein Visualisierungs- und Überwachungstool, das keine Änderungen an Ihrem Workflow vornehmen kann. Signalkarten stellen eine point-in-time Visualisierung Ihres Workflows dar. Für den Fall, dass Sie Teile Ihres Medien-Workflows hinzufügen, entfernen oder erheblich ändern, empfehlen wir Ihnen, die Signalmap erneut zu entdecken. Wenn Sie mit der Signalmap Monitoring-Ressourcen verknüpft haben, empfehlen wir Ihnen, die Überwachung nach dem Rediscovery-Prozess erneut einzusetzen.

Um eine Signalkarte wieder zu entdecken

1. Wählen Sie im Navigationsbereich der Workflow-Monitor-Konsole die Option Signalzuordnungen und dann die Signalkarte aus, mit der Sie arbeiten möchten.
2. Stellen Sie sicher, dass Sie sich in der Ansicht Signalmap konfigurieren befinden. Weitere Informationen zum Ändern von Ansichten finden Sie unter: [Signalkarten anzeigen](#)
3. Wählen Sie oben rechts auf der Signalzuordnungsseite das Dropdownmenü Aktionen aus. Wählen Sie Wiederentdecken aus.
4. Ihnen wird der Wiederentdeckungsbildschirm angezeigt. Wählen Sie eine Ressource aus, die Teil des Workflows ist, den Sie gerade wiederentdecken. Wählen Sie die Schaltfläche „Wiederentdecken“.
5. Die Signalkarte wird gemäß dem aktuellen Arbeitsablauf neu erstellt. Wenn Sie Monitoring-Ressourcen erneut einsetzen müssen, bleiben Sie auf der Seite dieser Signalübersicht. Alle zuvor angehängten Überwachungsvorlagen bleiben angehängt, müssen jedoch erneut bereitgestellt werden.

Um Überwachungsvorlagen nach einer erneuten Erkennung der Signalmap erneut bereitzustellen

1. Nach der Wiederentdeckung werden Sie zur aktualisierten Signalkarte weitergeleitet. Um die Überwachungsvorlagen erneut bereitzustellen, wählen Sie im Dropdownmenü Aktionen die Option Monitor bereitstellen aus.
2. Sie werden aufgefordert, die Bereitstellung zu bestätigen, und es wird die Anzahl der EventBridge Ressourcen angezeigt CloudWatch , die erstellt werden sollen. Wenn Sie fortfahren möchten, wählen Sie Bereitstellen aus.
3. Der Status der Bereitstellung wird neben dem Namen der Signalkarte angezeigt. Nach einigen Augenblicken der Erstellung und Bereitstellung der Ressourcen beginnt die Überwachung der Signalkarte.

Löschen der Signallampe Ihres AWS Medien-Workflows

Wenn Sie eine Signalkarte nicht mehr benötigen, können Sie sie löschen. Wenn Sie Monitoring-Vorlagen auf der Signal-Map bereitgestellt haben, werden Sie beim Löschen aufgefordert, alle CloudWatch EventBridge Ressourcen zu löschen, die auf dieser Signal-Map bereitgestellt wurden. Das Löschen der bereitgestellten Ressourcen hat keine Auswirkungen auf die Vorlagen, mit denen sie erstellt wurden. Durch das Löschen von Ressourcen soll sichergestellt werden, dass Sie nicht über EventBridge Ressourcen verfügen CloudWatch , die bereitgestellt, aber nicht verwendet werden.

Um eine Signalkarte zu löschen

1. Wählen Sie im Navigationsbereich der Workflow-Monitor-Konsole die Option Signalzuordnungen aus und klicken Sie auf das Optionsfeld neben der Signalmap, die Sie löschen möchten.
2. Wählen Sie die Schaltfläche Löschen aus. Sie werden aufgefordert, das Löschen der Monitoring-Ressourcen zu bestätigen. Wählen Sie Löschen aus, um mit dem Löschen der Überwachungsressourcen zu beginnen.
3. In der Spalte Bereitstellung überwachen wird der aktuelle Status angezeigt. Wenn sich der Status in DELETE_COMPLETE geändert hat, klicken Sie erneut auf die Schaltfläche Löschen.
4. Sie werden aufgefordert, das Löschen der Signalkarte zu bestätigen. Wählen Sie Löschen, um fortzufahren und die Signalkarte zu löschen.

Workflow-Monitor-Kontingente

Der folgende Abschnitt enthält Kontingente für Workflow-Monitor-Ressourcen. Jedes Kontingent wird „pro Konto“ berechnet. Wenn Sie ein Kontingent für Ihr Konto erhöhen müssen, können Sie die [AWS Servicekontingenten-Konsole](#) verwenden, um eine Erhöhung zu beantragen, sofern in der folgenden Tabelle nichts anderes angegeben ist.

Kontingente

Ressourcentyp	Kontingent
CloudWatch Gruppen von Alarmvorlagen	20
CloudWatch Alarm-Vorlagen	200
EventBridge Gruppen von Regelvorlagen	20
EventBridge Regelvorlagen	200
Signalkarten	30
Signalkarten: CloudWatch Alarmvorlagengruppen, die an eine einzelne Signalkarte angehängt sind	5 Sie können dieses Kontingent nicht erhöhen.
Signalzuordnungen: Gruppen von EventBridge Regelvorlagen, die an eine einzelne Signalmap angehängt sind	5 Sie können dieses Kontingent nicht erhöhen.

Verwenden Sie den Workflow-Monitor

Verwenden Sie die Bereiche Übersicht und Signalzuordnungen der Workflow-Monitor-Konsole, um den aktuellen Status der Workflows und alle zugehörigen Alarme, Metriken und Protokolle zu überprüfen.

Themen

- [Überblick über den Workflow-Monitor](#)
- [Übersicht über Protokolle und Metriken für den Workflow-Monitor](#)
- [Verwendung von Signalkarten im Workflow-Monitor](#)

Überblick über den Workflow-Monitor

Der Bereich „Übersicht“ der Workflow-Monitor-Konsole ist ein Dashboard, das at-a-glance Informationen zu Ihren Signalzuordnungen enthält. Im Übersichtsbereich können Sie den aktuellen Status der Überwachung jeder Signalmap sowie die CloudWatch Metriken und alle zugehörigen CloudWatch Protokolle einsehen. Sie können eine beliebige Signalmap auswählen, um auf die Konsolenseite der Signal Maps zu gelangen.

Filterung im Überblick

Mithilfe der Suchleiste im Übersichtsbereich können Sie die Liste der Signalzuordnungen anhand kontextsensitiver Einschränkungen filtern. Nachdem Sie die Suchleiste ausgewählt haben, wird Ihnen eine Liste mit Eigenschaften angezeigt, nach denen Sie filtern können. Wenn Sie eine Eigenschaft auswählen, werden Operatoren wie „Ist gleich“, „Enthält“, „Entspricht nicht“ und „Enthält nicht“ angezeigt. Wenn Sie einen Operator auswählen, wird eine Liste mit Ressourcen aus dem ausgewählten Eigenschaftstyp erstellt. Wenn Sie eine dieser Ressourcen auswählen, werden in der Signalzuordnungsliste nur Signalzuordnungen angezeigt, die der von Ihnen definierten Einschränkung entsprechen.

Übersicht über Protokolle und Metriken für den Workflow-Monitor

Um CloudWatch Metriken und Protokolle für eine Signal-Map anzuzeigen, wählen Sie das Optionsfeld neben dem Namen der Signal-Map aus. Unter der Liste der Signalzuordnungen wird eine Oberfläche mit Registerkarten für Metriken und Protokolle angezeigt.

CloudWatch Metriken

CloudWatch Die Metriken für die ausgewählte Signalmap sind kontextsensitiv und zeigen nur Metriken an, die mit den in diesem Signalkarten-Workflow verwendeten Diensten verknüpft sind. Sie können die Metriktools auf dem Bildschirm verwenden, um die angezeigten Metrikperioden und Zeitbereiche anzupassen.

CloudWatch Protokolle

Wenn Sie der Signalkarte eine CloudWatch Protokollgruppe zugeordnet haben, wird diese Gruppe hier angezeigt.

Verwendung von Signalkarten im Workflow-Monitor

Im Übersichtsbereich der Konsole können Sie eine bestimmte Signal-Map auswählen, um weitere Informationen zu dieser Signal-Map und den zugehörigen Monitoring-Ressourcen zu erhalten.

Nachdem Sie eine Signal-Map ausgewählt haben, werden Ihnen die Signal-Map und eine Reihe von Abschnitten mit Tabs angezeigt, die weitere Informationen enthalten:

- CloudWatch Alarme
- EventBridge Regeln
- AWS Elementare Warnungen
- Metriken
- Logs (Protokolle)
- Grundlegende Einzelheiten

In der Signalkarte navigieren

Eine Signalmap enthält Knoten für jede unterstützte AWS-Ressource, die vom Workflow-Monitor erkannt wurde. Bestimmte Ressourcen, wie MediaLive Kanäle und MediaPackage Endpunkte, können Vorschaubilder des Inhalts anzeigen, sofern Vorschaubilder verfügbar sind.

Wenn Sie einen Ressourcenknoten auswählen und im Dropdownmenü Aktionen die Option Ausgewählte Ressourcendetails anzeigen auswählen, gelangen Sie zur Detailseite des zugehörigen Dienstes. Wenn Sie beispielsweise einen MediaLive Kanal auswählen und Ausgewählte Ressourcendetails anzeigen auswählen, wird die Detailseite der MediaLive Konsole für diesen Kanal geöffnet.

Wenn Sie einen Ressourcenknoten auswählen, wird die Liste der aktiven Alarme nur auf diesen Knoten gefiltert. Wenn Sie im aktiven Alarm den Ziel-ARN der Ressource auswählen, werden Sie zur Detailseite des zugehörigen Dienstes weitergeleitet, auf der die ausgewählte Ressource geöffnet ist.

Kontingente in AWS Elemental MediaTailor

MediaTailor Anfragen für Ressourcen und Betriebsabläufe unterliegen den folgenden Kontingenten (früher als „Beschränkungen“ bezeichnet).

Sie können den AWS Service Quotas verwenden, um Kontingente einzusehen und Kontingenterhöhungen für MediaTailor viele andere AWS Dienste zu beantragen. Weitere Informationen zu diesem Service finden Sie im [Benutzerhandbuch für Service Quotas](#).

Kontingente für das Einfügen von Anzeigen

In der folgenden Tabelle werden die Kontingente für das Einfügen von AWS Elemental MediaTailor Anzeigen beschrieben. Sofern nicht anders angegeben, sind die Kontingente nicht anpassbar.

Name	Default quota value (Standardkontingentwert)	Beschreibung	
Länge für Ad Decision Server (Entscheidungsserver für Werbung, ADS)	25,000	Die maximale Anzahl von Zeichen in einer Spezifikation des Ad Decision Server (Entscheidungsserver für Werbung, ADS).	
Umleitungen für Ad Decision Server (Entscheidungsserver für Werbung, ADS)	5	Die maximale Tiefe der Weiterleitungen, die in VAST-Wrap per-Tags MediaTailor folgt. MediaTailor gibt auf, wenn es zusätzliche Weiterleitungen gibt.	
Zeitüberschreitung für Ad Decision Server	3	Die maximale Anzahl von Sekunden, die MediaTailor warten,	

Name	Default quota value (Standardkontingen twert)	Beschreibung	
(Entscheidungsserver für Werbung, ADS)		bis das Timeout bei einer offenen Verbindung zu einem Ad Decision Server (ADS) überschri tten wird. Wenn bei einer Verbindung das Zeitlimit überschri tten MediaTailor wird, weil der ADS nicht reagiert, kann die verfügbare Anzeige nicht mit Werbung gefüllt werden.	
Werbeeinblendungs- Anforderungen	10.000	Die maximale Anzahl an Anfragen pro Sekunde für personali sierte Manifeste bei serverseitiger Anzeigeneinblendun g. Die Anzeigene infügung verarbeitet eingehende Anfragen nach Manifesten, Sitzungsinitialisierung, Tracking-Daten und Anzeigensegmenten. Dieses Kontingent ist einstellbar.	

Name	Default quota value (Standardkontingen twert)	Beschreibung	
Konfigurationen	1.000	Die maximal zulässige Anzahl von Konfigurationen MediaTailor .	
Länge für Ursprungs-Server der Inhalte	512	Die maximale Anzahl von Zeichen in einer Spezifikation des Ursprungs-Servers der Inhalte.	
Zeitüberschreitung für Ursprungs-Server der Inhalte	2	Die maximale Anzahl von Sekunden, die bei der Anforderung von Vorlagenmanifesten vor einem Timeout bei einer offenen Verbindung zum Inhaltsursprungsserver MediaTailor gewartet wird. Zeitüberschreitung en generieren HTTP 504 (GatewayTimeoutException) -Antwort-Fehler.	

Name	Default quota value (Standardkontingen- twert)	Beschreibung	
Manifestgröße	2	Die maximale Größe aller originalen Wiedergabemanifeste in MB. Um sicherzustellen, dass das Kontingent nicht überschritten wird, können Sie Ihre Eingabemanifeste mit gzip in MediaTailor komprimieren.	
Verpackungskonfigurationen	5	Die maximale Anzahl von Paketkonfigurationen pro Quelle (ob Live- oder Video-on-Demand).	
Zeitpläne vorab abrufen	25	Die maximale Anzahl aktiver Prefetch-Zeitpläne pro Wiedergabekonfiguration. Abgelaufene Prefetch-Zeitpläne werden nicht auf dieses Limit angerechnet.	

Name	Default quota value (Standardkontingen twert)	Beschreibung	
Timeout für serversei tige Reporting- Beacon-Anfragen	3 Sekunden	Die maximale Anzahl von Sekunden, die vor einem MediaTailor Timeout bei einer offenen Verbindung zum Server gewartet wird, wenn ein Beacon für serverseitige Berichte ausgelöst wird. Wenn bei einer Verbindung das Timeout überschri tten MediaTailor wird, der Beacon nicht ausgelöst werden kann und der Dienst eine ERROR_FIR ING_BEACO N_FAILED Meldung unter MediaTailor/ AdDecisionServe rInteractionsprotokoll iert, melden Sie sich an. CloudWatch	

Name	Default quota value (Standardkontingentwert)	Beschreibung	
Sitzungsablauf	Ein 10-faches der Manifestdauer	Die maximale Zeit, während der eine MediaTailor Sitzung inaktiv bleiben kann, bevor die Sitzung beendet wird. Zu Sitzungsaktivitäten können Player-Anforderungen oder Kontaktversuche durch den Ursprungs-Server zählen. Wenn die Sitzung abläuft, wird ein HTTP 400 (Bad Request) Antwortfehler MediaTailor zurückgegeben.	

Kontingente für die Kanalmontage

In der folgenden Tabelle werden die Kontingente für die AWS Elemental MediaTailor Kanalmontage beschrieben. Sofern nicht anders angegeben, sind die Kontingente [anpassbar](#).

Name	Default quota value (Standardkontingentwert)	Beschreibung	
Anfragen nach Channel-Manifests, pro Konto	400	Die maximale Anzahl von ausgehenden Manifestanfragen pro Sekunde für alle	

Name	Default quota value (Standardkontingen twert)	Beschreibung	
		Channel Assembly- Kanäle eines Kontos.	
Kanalmanifestanfra gen pro Kanal	50	Die maximale Anzahl von ausgehenden Manifestanfragen pro Sekunde für einen einzelnen Channel Assembly-Kanal.	
Kanal-Ausgaben	5	Die maximale Anzahl von Ausgaben pro Kanal.	
Kanäle pro Konto	100	Die maximale Anzahl von Kanälen pro Konto.	
Live-Quellen	50	Die maximale Anzahl von Live-Quellen für den Quellstandort.	
Programme pro Kanal	400	Die maximale Anzahl der Programme pro Kanal.	
Konfigurationen für die Segmentzu stellung	5	Die maximale Anzahl von Konfigurationen der Segmentbe reitstellung pro Quellstandort.	
Quellstandorte	50	Die maximale Anzahl von Quellstandorten pro Konto.	

Name	Default quota value (Standardkontingen twert)	Beschreibung	
VOD-Quellen	1.000	Die maximale Anzahl von Video-on-Demand-Quellen (VOD) für den Quellstandort.	

In der folgenden Tabelle werden die Drosselungsgrenzen für die Kanalmontage beschrieben. AWS Elemental MediaTailor [Sofern nicht anders angegeben, sind die Kontingente anpassbar.](#)

Name	transactions-per-secondStandard-Höchstlimit	Beschreibung	
ConfigureLogsForChannel	1	Konfigurieren Sie Protokolle für den Kanal.	
CreateChannel	1	Erstellen Sie einen Kanal.	
CreateLiveSource	1	Erstellen Sie eine Live-Quelle.	
CreateProgram	3	Erstellen Sie ein Programm.	
CreateSourceLocation	1	Erstellen Sie einen Quellspeicherort.	
CreateVodSource	1	Erstellen Sie eine VOD-Quelle.	
DeleteChannel	1	Löscht einen Kanal.	

Name	transactions-per-secondStandard-Höchstlimit	Beschreibung	
DeleteChannelPolicy	1	Löscht eine Kanalrichtlinie.	
DeleteLiveSource	1	Löscht eine Live-Quelle.	
DeleteProgram	3	Löscht ein Programm.	
DeleteSourceLocation	1	Löscht einen Quellspeicherort.	
DeleteVodSource	1	Löscht eine VOD-Quelle.	
DescribeChannel	5	Beschreiben Sie einen Kanal.	
DescribeLiveSource	5	Beschreiben Sie eine Live-Quelle.	
DescribeProgram	5	Beschreiben Sie ein Programm.	
DescribeSourceLocation	5	Beschreiben Sie einen Quellstandort.	
DescribeVodSource	5	Beschreiben Sie eine VOD-Quelle.	
GetChannelPolicy	5	Holen Sie sich eine Kanalrichtlinie.	
GetChannelSchedule	5	Holen Sie sich einen Kanalplan.	

Name	transactions-per-secondStandard-Höchstlimit	Beschreibung	
ListAlerts	5	Benachrichtigungen auflisten.	
ListChannels	5	Kanäle auflisten.	
ListLiveSources	5	Live-Quellen auflisten.	
ListPrograms	5	Programme auflisten.	
ListSourceLocations	5	Quellorte auflisten.	
ListTagsForResource	5	Listet Tags für eine Ressource auf	
ListVodSources	5	VOD-Quellen auflisten .	
PutChannelPolicy	3	Legen Sie eine Kanalrichtlinie fest.	
StartChannel	1	Starte einen Kanal.	
StopChannel	1	Stoppen Sie einen Kanal.	
TagResource	1	Versieht eine Ressource mit Tags	
UntagResource	1	Entfernt Tags von einer Ressource	
UpdateChannel	1	Aktualisiere einen Kanal.	
UpdateLiveSource	1	Aktualisiere eine Live-Quelle.	

Name	transactions-per-secondStandard-Höchstlimit	Beschreibung	
UpdateProgram	1	Aktualisieren Sie ein Programm.	
UpdateSourceLocation	1	Aktualisieren Sie einen Quellspeicherort.	
UpdateVodSource	1	Aktualisieren Sie eine VOD-Quelle.	

AWS Elemental MediaTailor Ressourcen

Die folgende Tabelle enthält verwandte Ressourcen, die für die Arbeit mit AWS Elemental MediaTailor nützlich sind.

Ressource	Beschreibung
SCTE-Standard: SCTE 35	Das SCTE-Standarddokument für SCTE35
Kurse und Workshops	Links zu rollenbasierten und speziellen Kursen sowie Übungen im Selbststudium zur Verbesserung Ihrer AWS-Kompetenzen und für praktische Erfahrung.
AWS-Entwicklertools	Links zu Entwicklertools SDKs, IDE-Toolkits und Befehlszeilentools für die Entwicklung und Verwaltung von AWS-Anwendungen.
AWS-Whitepapers	Links zu einer umfangreichen Liste technischer AWS-Whitepaper zu Themen wie Architektur, Sicherheit und Wirtschaftlichkeit. Diese Whitepaper wurden von AWS-Lösungsarchitekten und anderen technischen Experten verfasst.
AWS-Supportcenter	Die Zentrale zum Anlegen und Verwalten Ihrer AWS-Support-Fälle. Enthält auch Links zu anderen hilfreichen Ressourcen wie Foren, technischen Informationen FAQs, Service Health Status und AWS Trusted Advisor.
AWS Support	Die wichtigste Webseite mit Informationen zu AWS Support, einem Support-Kanal mit schnellen Reaktionszeiten one-on-one, der Sie beim Erstellen und Ausführen von Anwendungen in der Cloud unterstützt.

Ressource	Beschreibung
Kontakt	Eine zentrale Anlaufstelle bei Fragen zu AWS-Abrechnungen, Konten, Vorfällen, Missbrauch und anderen Problemen.
Nutzungsbedingungen der AWS-Website	Detaillierte Informationen zu unseren Copyright - und Markenbestimmungen, Ihrem Konto, den Lizenzen und anderen Themen.

Dokumenthistorie für AWS Elemental MediaTailor

In der folgenden Tabelle werden wichtige Änderungen an dieser Dokumentation beschrieben.

Änderung	Beschreibung	Datum
Wiederkehrender Prefetch hinzugefügt	Die Abschnitte wurden aktualisiert und enthalten nun Informationen zu wiederkehrenden Prefetch-Zeitplänen.	28. April 2025
Es wurden konfigurationsspezifische Informationen zur Protokollfilterung hinzugefügt	MediaTailor Es wurden Informationen zum Filtern von Protokollen hinzugefügt, die für eine Konfiguration ausgegeben werden.	31. März 2025
Informationen zu verkauften Protokollen wurden hinzugefügt	Es wurde ein neuer Abschnitt über die Verwendung von versendeten Protokollen für die Veröffentlichung von Protokollen hinzugefügt, die ausgegeben.. MediaTailor	4. Februar 2025
Übersicht über vorkonditionierte Anzeigen hinzugefügt	Es wurde ein neuer Abschnitt hinzugefügt, in dem beschrieben wird, wie die Anzeigeneinblendung MediaTailor verwaltet wird, wenn Sie vorkonditionierte Anzeigen verwenden.	30. Januar 2025
Anzeigenkonditionierung hinzugefügt	Die Einstellung zur Konditionierung von Streaming-Mediendateien wurde hinzugefügt.	30. Januar 2025

<u>Protokolltypen wurden hinzugefügt</u>	Es wurde ein neuer Abschnitt hinzugefügt, der die Protokolltypen auflistet, die MediaTailor ausgegeben werden.	15. Januar 2025
<u>Informationen zur Protokollfilterung wurden hinzugefügt</u>	MediaTailor Es wurde ein neuer Abschnitt über das Filtern von ausgegebenen Protokollen hinzugefügt.	15. Januar 2025
<u>Unterstützte Formatierung von Abfrageparametern hinzugefügt</u>	Es wurden Abschnitte für die Formatierung von Manifest-Abfrageparametern und ADS-Abfrageparametern hinzugefügt.	2. Januar 2025
<u>Informationen zur Integration für Google Ad Manager</u>	Es wurde ein Abschnitt zur Integration MediaTailor mit Google Ad Manager von der Server- und Clientseite hinzugefügt.	25. November 2024
<u>Variable aktualisiert</u>	Geändert breakabilityStartTime zuavailabilityStartTime .	6. Mai 2024
<u>Details zur Personalisierung hinzugefügt</u>	Das Insertion Mode Personalisierungsdetail wurde hinzugefügt.	6. Mai 2024
<u>Regeln des Programms</u>	Neue Inhalte zu den Programmregeln hinzugefügt.	25. April 2024
<u>Der Inhalt zum Erstellen von Kanälen wurde aktualisiert</u>	Es wurden Informationen zu Programmregeln beim Erstellen von Kanälen hinzugefügt.	20. April 2024

Der Inhalt zum Hinzufügen von Programmen wurde aktualisiert	Es wurden Informationen zu Programmregeln beim Hinzufügen von Programmen hinzugefügt.	20. April 2024
Workflow-Monitor	Analysieren Sie AWS Mediendienste und erstellen Sie Signalkarten, Visualisierungen des Medienworkflows, zwischen diesen Diensten. Verwenden Sie die Signalzuordnungen, um mithilfe CloudWatch von, und Überwachungsalarme und -benachrichtigungen zu generieren. EventBridge AWS CloudFormation	11. April 2024
AlternateMedia und das As Run Log	Es wurde ein Hinweis hinzugefügt, wie Alternate Media sich das As Run Log auswirkt.	28. Februar 2024
Zeitversetztes Betrachten	MediaTailor Kanäle unterstützen die zeitversetzte Anzeige von Inhalten, die bis zu 6 Stunden alt sind.	27. Dezember 2023
Die Manifest-Einstellungen wurden aktualisiert	Je nach ausgewähltem Anzeigen-Markup-Typ wurden Informationen zu Passthrough-Tags hinzugefügt.	28. November 2023

<u>SCTE-35-Nachrichten für Werbeunterbrechungen</u>	Es wurden Informationen zu eingefügten SCTE-35-Tags für und zu Anzeigenmarkup-Typen hinzugefügt. <code>DateRange</code> . <code>Scte35 Enhanced</code>	28. November 2023
<u>Schlüssel-Wert-Paare für den Markup-Typ „Werbung“ Enhanced Scte35</u>	Es wurden Informationen zum MediaTailor Umgang mit übermittelten Schlüssel-Wert-Paaren für den Markup-Typ „Werbung“ hinzugefügt. <code>Enhanced Scte35</code>	28. November 2023
<u>VOD-Quellen und Möglichkeiten</u>	MediaTailor kann jetzt automatisch Werbemöglichkeiten in VOD-Quellen erkennen.	06. Oktober 2023
<u>Neuer SigV4-Authentifizierungstyp Autodetect</u>	MediaTailor unterstützt jetzt den <code>AUTODETECT_SIGV4</code> Zugriffstyp.	18. August 2023
<u>Der clientseitige Tracking-Inhalt wurde aktualisiert</u>	Der Inhalt der clientseitigen Nachverfolgung wurde aktualisiert und enthält nun zusätzliche Informationen.	12. August 2023
<u>Einrichtung MediaTailor und MediaPackage Workflows zur Verwendung von Live-Quellen</u>	Es wurden Informationen zur Einrichtung, zu allgemeinen Anforderungen und zum Verhalten bei MediaTailor der Verwendung von Live-Quellen in AWS Elemental MediaPackage Workflows hinzugefügt.	24. Mai 2023
<u>Dokumentation zu den Manifest-Abfrageparametern</u>	Es wurde ein Abschnitt hinzugefügt, der die Manifest-Abfrageparameter beschreibt.	26. April 2023

Dokumentation zu Overlay-Anzeigen	Es wurde ein Abschnitt hinzugefügt, der Overlay-Anzeigen beschreibt.	24. April 2023
Dekoration der Anzeigen-ID in der Dokumentation zu den Manifesten	Es wurde ein Abschnitt hinzugefügt, in dem die Dekoration von Anzeigen-IDs in Manifesten beschrieben wird.	24. April 2023
Der AFTER_LIVE_EDGE-Unterdrückungsmodus wurde hinzugefügt	AFTER_LIVE_EDGE Der Werbeunterdrückungsmodus ist jetzt zusätzlich zum Modus verfügbar. BEFORE_LIVE_EDGE	21. Februar 2023
Neues As-Run-Protokoll	Neues Thema im As Run-Protokoll.	19. Januar 2023
Updates zu den bewährten Methoden für IAM	Aktualisierung des Leitfadens zur Ausrichtung an bewährten IAM-Methoden. Weitere Informationen finden Sie unter Bewährte IAM-Methoden .	27. Dezember 2022
Updates zu bewährten IAM-Praktiken	Aktualisierung des Leitfadens zur Ausrichtung an bewährten IAM-Methoden. Weitere Informationen finden Sie unter Bewährte IAM-Methoden .	27. Dezember 2022
Inhalt der Kontingente aktualisiert	Die Informationen zu den Kontingenten wurden aktualisiert und neu organisiert.	13. September 2022

<u>Kontingente für Kanalzusammenstellungen wurden hinzugefügt und korrigiert</u>	Es wurden Kontingente für Live-Quellen, Konfigurationen für die Segmentzuweisung, Manifestanfragen und Kanaltransaktionen im MediaTailor Channel Assembly Service hinzugefügt.	13. September 2022
<u>Kontingente für Channel-Assembly hinzugefügt und korrigiert</u>	Es wurden Kontingente für Protokollierung, Kanal, Live-Quellen, Programme, Quelladressen und Kanalrichtlinien im MediaTailor Channel Assembly Service hinzugefügt.	11. September 2022
<u>Neue Tabelle mit Warnmeldungen zur Kanalbaugruppe</u>	Sie können jetzt eine Tabelle sehen, in der die Warnmeldungen zur Kanalbaugruppe erklärt werden.	01. September 2022
<u>Neue CloudWatch Amazon-Metriken</u>	Neue CloudWatch Metriken hinzugefügt.	26. Juni 2022
<u>Anzeige ruft Themen auf</u>	Link zur Prefetch-Unterstützung für VAST-Antworten hinzugefügt.	25. Mai 2022

[Neue ADS-Anforderungsvariablen](#)

MediaTailor unterstützt jetzt diese zusätzlichen SCTE-35-Variablen in ADS-Anfragen: `scte.segmentation_type_id`, `scte.available_expected`, `scte.delivery_not_restricted_flag`, `scte.segment_num`, `scte.sub_segment_num`, `scte.segments_expected`, `scte.sub_segments_expected`, `scte.device_restrictions`, `scte.no_regional_blackout_flag`, `scte.archive_allowed_flag` und `scte.segmentation_event_id`.

27. April 2022

[Neues Thema zu verwalteten IAM-Richtlinien](#)

Es wurden zwei neue verwaltete Richtlinien für MediaTailor hinzugefügt.

24. November 2021

[Neue von AWSElementalMediaTailorReadOnly verwaltete Richtlinie](#)

Es wurde eine neue AWS verwaltete Richtlinie hinzugefügt, die Berechtigungen gewährt, die den schreibgeschützten Zugriff auf Ressourcen ermöglichen MediaTailor .

10. November 2021

<u>Neue von AWSElementalMediaTailorFullAccess verwaltete Richtlinie</u>	Es wurde eine neue AWS verwaltete Richtlinie hinzugefügt, die vollen Zugriff auf Ressourcen ermöglicht MediaTailor.	10. November 2021
<u>Neues verwirrtes stellvertretendes Thema</u>	Es wurde ein Thema hinzugefügt, in dem erklärt wird, wie das Problem mit dem verwirrten Stellvertreter verhindert werden kann.	4. November 2021
<u>Das Anzeigenthema wird vorab abgerufen</u>	MediaTailor kann jetzt Anzeigen für Werbeunterbrechungen vorab abrufen, bevor sie auftreten.	12. Oktober 2021
<u>Es wurden Konfigurationseinstellungen für die Protokollierung von Wiedergabekonfigurationen hinzugefügt</u>	Verwenden Sie die Konfigurationseinstellungen für die Protokollierung, um die Einstellungen für die Wiedergabekonfigurationsprotokolle zu steuern.	28. September 2021
<u>SCTE-35-Meldungen für Werbeunterbrechungen</u>	Es wurden Informationen zur Verwendung von <code>segmentation_descriptor</code> Nachrichten zusammen mit <code>time_signal</code> Nachrichten hinzugefügt.	1. September 2021
<u>Neuer linearer Wiedergabemodus</u>	Ein neuer linearer Wiedergabemodus wurde hinzugefügt.	1. September 2021

Neuer absoluter Übergangstyp	Es wurde Unterstützung für absolute Übergangstypen hinzugefügt, mit denen Sie eine Wanduhr-Startzeit für Ihr Programm auf linearen Kanälen festlegen können.	1. September 2021
Neues Thema „Benachrichtigungen zur Kanalmontage“	Sie können jetzt Ihre Ressourcen für die Kanalzusammenstellung mithilfe von MediaTailor Benachrichtigungen überwachen. Wenn ein Problem oder ein potenzielles Problem mit Ihren Ressourcen für die Kanalmontage auftritt, werden Warnmeldungen MediaTailor generiert.	14. Juli 2021
Die Kontingente für Kanalzusammenstellungen für ausgehende Kanalanfragen wurden korrigiert	Die Kontingente für ausgehende Kanalanfragen im MediaTailor Channel-Assembly-Service wurden korrigiert.	29. Juni 2021
Neuer Authentifizierungstyp für den Quellstandort	MediaTailor unterstützt jetzt die Secrets Manager Manager-Zugriffstoken-Authentifizierung.	16. Juni 2021
Neue Tier-Informationen	Es wurden Informationen zu den Modi und Quelltypen hinzugefügt, die jede Stufe unterstützt.	13. Juni 2021

Neue Informationen zum Quellentyp	Für Standardkanäle wurden Informationen über die Art der Quelle hinzugefügt, die das Programm wiedergibt.	13. Juni 2021
Neue Dokumentation MediaTailor zu Live-Quellen	Eine Live-Quelle stellt einen einzelnen Live-Stream dar, den Sie zu Ihrem Quellstandort hinzufügen. Nachdem du deinen Kanal erstellt hast, kannst du Live-Quellen zu deinem Quellstandort hinzufügen und jede Live-Quelle einem Programm zuordnen.	13. Juni 2021
Support für zusätzliche UPID-Typen	MediaTailor unterstützt jetzt die UPID-Typen ADS-Information (0xE) und benutzerdefinierte Segmentierung (0x1).	15. April 2021
Neue dynamische UPID-Variablen für die Segmentierung	Es gibt drei neue dynamische Variablen: <code>scte.segmentation_upid.assetId</code> , <code>scte.segmentation_upid.cueData.key</code> , und <code>scte.segmentation_upid.cueData.value</code> . Diese Variablen werden in Verbindung mit dem UPID-Typ der MPU-Segmentierung (0xC) für Podbuster-Workflows verwendet.	15. April 2021
Neue Servicebeschreibung für die Kanalmontage	Es wurden Informationen zum neuen Kanalmontage-Service hinzugefügt.	11. März 2021

Neue Servicedokumentation für die MediaTailor Kanalmontage	Channel Assembly ist ein neuer Dienst, der nur auf Manifest beschränkt ist und es Ihnen ermöglicht, lineare Streaming-Kanäle unter Verwendung Ihrer vorhandenen Video-on-Demand-Inhalte (VOD) zu erstellen.	11. März 2021
Kontingente für die Kanalzusammenstellung hinzugefügt	Kontingente für den neuen MediaTailor Channel-Assembly-Service hinzugefügt.	11. März 2021
Neue Bedingungen für die Kanalmontage	Es wurden Bedingungen hinzugefügt, die dem neuen Service zur Kanalmontage entsprechen.	10. März 2021
Unterstützung für die Kennzeichnung von Kanalbaugruppen	Unterstützung für das Taggen von Ressourcen zur Kanalmontage wurde hinzugefügt. AWS Elemental MediaTailor Kanäle SourceLocations, und VodSources unterstützen das Taggen.	9. März 2021
Neues Thema zu dynamischen Variablen	MediaTailor unterstützt jetzt dynamische Domänenvariablen.	25. Februar 2021
Optionale Alias-Einstellungen für die Konfiguration wurden hinzugefügt	Verwenden Sie Konfigurationsalias zusammen mit Domänenvariablen, um Domänen während der Sitzungsinitialisierung dynamisch zu konfigurieren.	25. Februar 2021

<u>Neue scte.segmentation_upid dynamische Anzeigenvariable</u>	Unterstützung für die dynamische Anzeigenvariable scte.segmentation_upid Sitzungsdaten wurde hinzugefügt.	5. Dezember 2020
<u>Neues Passthrough-Thema zur Anzeigenmarkierung</u>	Passthrough für Werbemerkierungen ist jetzt für HLS-Manifeste verfügbar.	29. Oktober 2020
<u>Die erweiterten Konfigurationseinstellungen wurden aktualisiert</u>	Ad Marker Passthrough ist eine neue erweiterte Einstellung für die Wiedergabekonfiguration.	14. Oktober 2020
<u>Neuer Debug-Log-Modus</u>	Neues Thema zum DEBUG-Logmodus.	14. August 2020
<u>Klarstellung zum EXT-X-CUE-OUT Dauerattribut für Stoßstangen</u>	Die Anforderungen an Bumpers wurden aktualisiert, sodass für HLS das duration Attribut für jedes Tag erforderlich ist. EXT-X-CUE-OUT	5. August 2020
<u>Neues Bumper-Thema</u>	Ein neues Bumper-Thema wurde hinzugefügt	27. Juli 2020
<u>Ad Suppression ist für DASH verfügbar</u>	Für DASH ist jetzt Werbeunterdrückung verfügbar. Die „Nur HLS“-Einschränkung wurde aus dem Thema der Werbeunterdrückung entfernt.	3. Juni 2020

<u>Aktualisieren Sie konsolenspezifische Namen</u>	Die konsolenspezifischen Namen wurden aktualisiert, um eine neuere Version der Konsolen-Benutzeroberfläche wiederzugeben.	1. Mai 2020
<u>Neue avail.index dynamische AnzeigenvARIABLE</u>	Hinzufügung der Unterstützung für die neue dynamische AnzeigenvARIABLE für avail.index -Sitzungsdaten.	13. März 2020
<u>Neu AdVerifications und Extensions Elemente</u>	Für die clientseitige Berichterstellung werden die AdVerifications - und Extensions -Elemente unterstützt.	10. März 2020
<u>Konfiguration des Schwellenwerts für die Personalisierung</u>	Ab sofort wird die optionale Konfiguration einer Personalisierungsschwelle unterstützt.	14. Februar 2020
<u>DASH VOD-Manifestationen</u>	Es wurde Unterstützung für DASH-VOD (Video on Demand)-Manifeste vom Ursprungs-Server mit mehreren Ausgabezeiträumen hinzugefügt.	23. Dezember 2019
<u>Konsolenunterstützung für den Transcode-Profilnamen</u>	Beschreibung für Transcodierungsprofilnamen in die Konfiguration eingefügt.	23. Dezember 2019
<u>Die Grenzwerttabellen wurden aktualisiert</u>	Aktualisierte Limits für ADS-Weiterleitungen und -Timeouts	18. Dezember 2019

Bewährte CDN-Praktiken	Es wurde ein Abschnitt über Best Practices für Content Distribution Network (CDN) für personalisierte Manifeste hinzugefügt.	13. Dezember 2019
Dokumentieren Sie Live-Pre-Roll-Verhaltensweisen	Es wurde ein Abschnitt zum Einfügen von Pre-Roll-Anzeigen hinzugefügt, in dem beschrieben wird, wie Live-Pre-Roll-Anzeigen mit funktionieren. AWS Elemental MediaTailor	26. November 2019
Support für Live-Pre-Roll-Anzeigen	Unterstützung für das Einfügen von Pre-Roll-Werbung am Anfang eines Live-Streams wurde hinzugefügt.	11. September 2019
Analysieren von ADS-Protokollen in Amazon CloudWatch Logs Insights	Es wurden Informationen zur Verwendung der AWS Elemental MediaTailor ADS-Logs und CloudWatch Logs Insights zur Analyse Ihrer MediaTailor Sitzungen hinzugefügt.	13. August 2019
Neues Kapitel bezüglich der Sicherheit	Es wurde ein Kapitel bezüglich der Sicherheit hinzugefügt, um die Abdeckung des Themas zu verbessern und zu standardisieren.	23. Mai 2019

<u>DASH-Manifeste für einzelne Perioden</u>	Es wurde Unterstützung für DASH-Manifeste mit einem einzigen Zeitraum vom Ursprungs-Server mit Manifest-Ausgabe mit mehreren Zeiträumen hinzugefügt.	4. April 2019
<u>Support für SCTE-35 UPIDs in der ADS-URL</u>	Es wurde Unterstützung für den Einschluss einer eindeutigen Programm-ID (UPID) in die URL des Ad Decision Server (Entscheidungsservers für Werbung, ADS) hinzugefügt. Dies ermöglicht dem ADS, zielgerichtete Werbung auf Programmebene in einem linearen Live-Stream bereitzustellen.	28. März 2019
<u>Die clientseitige Berichterstattung unterstützt Begleitanzeigen</u>	Für die clientseitige Berichterstattung enthält die Antwort auf die AWS Elemental MediaTailor Tracking-URL jetzt Metadaten für Begleitanzeigen.	28. März 2019
<u>Dokumentation zu HLS-Werbemarkern</u>	Es wurde ein Abschnitt zur Beschreibung unterstützter HLS-Werbemarkern hinzugefügt.	1. März 2019

Unterstützung für Tagging	Es wurde Unterstützung für das Tagging von Konfigurationsressourcen in AWS Elemental MediaTailor hinzugefügt. Tagging ermöglicht es Ihnen, Ihre AWS Ressourcen zu identifizieren und zu organisieren, den Zugriff darauf zu kontrollieren und Ihre AWS Kosten zu verfolgen.	14. Februar 2019
AWS CloudTrail Protokollierungsinformationen wurden hinzugefügt	Es wurde ein Thema CloudTrail zur Verwendung zum Protokollieren von Aktionen in der AWS Elemental MediaTailor API hinzugefügt.	11. Februar 2019
Abschnitt zu Wiedergabefehlern hinzugefügt	Es wurden Informationen über die Fehler hinzugefügt, die von MediaTailor während der Wiedergabe als Antwort auf Anforderungen von einem Player oder einem Netzwerk zur Bereitstellung von Inhalten (Content Delivery Network, CDN) zurückgegeben werden.	4. Februar 2019
Base64-codierte DASH-Binärdatei	Es wurde Unterstützung für die Bereitstellung von Splicing-Informationen in Manifesten im base64-kodierten binären Format innerhalb von <scte35:Signal> - und <scte35:Binary> -Markern hinzugefügt.	4. Januar 2019

DASH-Zeitsignal	Es wurde Unterstützung für die Bereitstellung von Splicing-Informationen in Manifesten innerhalb von <scte35:TimeSignal>-Markern hinzugefügt.	5. Dezember 2018
Unterstützung für DASH-Standorte	Unterstützung für das MPEG-DASH-Tag <Location> hinzugefügt.	4. Dezember 2018
DASH-Unterstützung	Unterstützung für MPEG-DASH-Manifeste hinzugefügt.	14. November 2018
Die Limittabellen wurden aktualisiert	Aktualisierte Limits für Konfigurationen und Manifest-Größe.	13. Oktober 2018
Neue und aktualisierte Metriken	Es wurden Metriken für den Ad Decision Server (Entscheidungsserver für Werbung, ADS) und Timeouts für den Ursprungs-Server hinzugefügt und die Fehlerdefinitionen des ADS und des Ursprungs-Servers so aktualisiert, dass Antworten mit Zeitüberschreitung eingeschlossen werden.	13. Oktober 2018
Bessere Dokumentation für Anwendungsfälle zur serverseitigen und clientseitigen Anzeigeneinfügung	Eingehendere Beschreibung und Beispiele, die auf die Verwendung von dynamischen Werbevariablen für serverseitige und clientseitige Werbeeinblendung eingehen.	1. Oktober 2018

Neue Regionen	Unterstützung für die Regionen PDX und FRA hinzugefügt.	18. Juli 2018
VAST/VPAID	Informationen über VAST und VPAID hinzugefügt.	16. März 2018
CloudWatch	Es wurden Informationen zu verfügbaren CloudWatch Metriken, Namespaces und Dimensionen hinzugefügt.	16. März 2018
Neue Regionen	Unterstützung für die Regionen Asien-Pazifik (Singapur), Asien-Pazifik (Sydney) und Asien-Pazifik (Tokio) hinzugefügt.	8. Februar 2018
CloudFront Standard-Vertriebspfade von Amazon	Es wurde die Liste der Pfade für die CloudFront Amazon-Distribution hinzugefügt, in der Anzeigen AWS Elemental MediaTailor gespeichert werden.	6. Februar 2018
Informationen zur IAM-Richtlinie	Spezifische IAM-Richtlinieninformationen für hinzugefügt. AWS Elemental MediaTailor Es wurden Anweisungen für die Erstellung von Nicht-Admin-Rollen mit eingeschränkten Berechtigungen hinzugefügt.	3. Januar 2018
Erste Veröffentlichung	Erste Version dieser Dokumentation.	27. November 2017

 Note

- Die AWS Mediendienste sind nicht für die Verwendung mit Anwendungen oder in Situationen konzipiert oder vorgesehen, in denen eine ausfallsichere Leistung erforderlich ist, wie z. B. bei der Betriebssicherheit, Navigations- oder Kommunikationssystemen, der Flugsicherung oder lebenserhaltenden Maschinen, in denen die Nichtverfügbarkeit, Unterbrechung oder der Ausfall der Dienste zu Tod, Personen-, Sach- oder Umweltschäden führen kann.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.