



User Guide

AWS Elemental MediaTailor



AWS Elemental MediaTailor: User Guide

Copyright © 2024 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 zum Einfügen von Anzeigen | 1 |
| Konzepte für die Kanalmontage | 2 |
| Wie funktioniert das Einfügen von MediaTailor Anzeigen | 3 |
| Zugehörige Services | 4 |
| Zugreifen MediaTailor | 5 |
| Preisgestaltung | 5 |
| Regionen | 5 |
| Einrichtung | 6 |
| Melden Sie sich an für eine AWS-Konto | 6 |
| Erstellen Sie einen Benutzer mit Administratorzugriff | 7 |
| Erste Schritte mit MediaTailor | 9 |
| Erste Schritte mit der MediaTailor Anzeigeneinfügung | 9 |
| Voraussetzungen | 10 |
| Schritt 1: Zugriff MediaTailor | 10 |
| Schritt 2: Bereite einen Stream vor | 10 |
| Schritt 3: Konfigurieren Sie die ADS Anfrage URL - und Abfrageparameter | 12 |
| Schritt 4: Erstellen einer Konfiguration | 13 |
| Schritt 5: Testen Sie die Konfiguration | 15 |
| Schritt 6: Senden Sie die Wiedergabeanforderung an MediaTailor | 17 |
| (Optional) Schritt 7: Aktivität überwachen | 18 |
| Schritt 8: Bereinigen | 19 |
| Erste Schritte mit der MediaTailor Kanalmontage | 20 |
| Voraussetzungen | 21 |
| Schritt 1: Erstellen Sie einen Quellspeicherort | 21 |
| Schritt 2: Fügen Sie VOD Quellen zu Ihrem Quellort hinzu | 22 |
| Schritt 3: Erstellen Sie einen Kanal | 23 |
| Schritt 4: Füge Programme zum Zeitplan deines Kanals hinzu | 25 |
| Schritt 5 (optional): Verwenden Sie diese Option MediaTailor , um personalisierte Anzeigen in Ihren Stream einzufügen | 27 |
| Schritt 6: Starte deinen Kanal | 28 |
| Schritt 7: Teste deinen Kanal | 28 |
| Schritt 8: Bereinigen | 28 |

| | |
|--|-----|
| Konfiguration MediaTailor als Ihr Dienst zur Anzeigeneinfügung | 29 |
| Unterstützte Audio- und Videocodecs | 29 |
| VAST, und Anforderungen für VMAP Werbeserver VPAID | 30 |
| VASTAnforderungen | 30 |
| VPAIDAnforderungen | 32 |
| Mit MediaTailor Konfigurationen arbeiten | 33 |
| Konfiguration erstellen | 33 |
| Konfiguration anzeigen | 40 |
| Eine Konfiguration bearbeiten | 41 |
| Eine Konfiguration löschen | 41 |
| Anpassen des Verhaltens von Werbeunterbrechungen | 42 |
| Die Unterdrückung von Werbeunterbrechungen konfigurieren | 42 |
| Integration einer Inhaltsquelle | 66 |
| Anforderungen an die Eingabequelle | 67 |
| Eine HLS Quelle integrieren | 67 |
| Integration einer MPEG DASH -Quelle | 76 |
| Overlay-Anzeigen | 138 |
| Voraussetzungen | 139 |
| Erste Schritte | 139 |
| Protokollierung und Metriken | 153 |
| Fakturierung | 155 |
| ID-Dekoration hinzufügen | 156 |
| Status der Sitzung | 157 |
| Manifeste und Einfügen von Anzeigenmetadaten | 158 |
| Interaktionen mit dem Entscheidungsserver (ADS) anzeigen | 182 |
| Clientseitiges Tracking API | 183 |
| Daten zur Berichterstattung und Nachverfolgung | 185 |
| Serverseitiges Tracking | 185 |
| Kundenseitiges Tracking | 186 |
| Verwendung dynamischer Anzeigenvariablen | 282 |
| Übergeben von Parametern an ADS | 283 |
| Fortgeschrittene Nutzung | 285 |
| Verwenden von Domänenvariablen | 286 |
| Verwenden von Sitzungsvariablen | 289 |
| Verwendung von Spielervariablen | 301 |
| Arbeiten mit CDNs | 304 |

| | |
|---|-----|
| Integration eines CDN | 304 |
| Wie MediaTailor geht man mit BaseURLs für DASH um | 309 |
| Bewährte Methoden für CDN | 309 |
| Das Verhalten bei der MediaTailor Anzeigeneinblendung verstehen | 310 |
| Verhalten beim Zusammenfügen von Anzeigen bei VOD | 310 |
| Verhalten beim Zusammenfügen von Live-Anzeigen | 313 |
| Grundlegendes zur MediaTailor servergesteuerten Anzeigeneinfügung | 315 |
| In der Wiedergabekonfiguration aktivieren | 315 |
| Erstellen Sie eine servergeführte Sitzung | 315 |
| Wird verwendet MediaTailor , um linear zusammengestellte Streams zu erstellen | 318 |
| Mit Quellstandorten arbeiten | 318 |
| Einen Quellspeicherort erstellen | 319 |
| Konfiguration der Authentifizierung für Ihren Quellstandort | 321 |
| Mit VOD Quellen arbeiten | 334 |
| Mit Live-Quellen arbeiten | 338 |
| Verwenden von Paketkonfigurationen | 343 |
| Zwischenspeichern von Manifesten | 343 |
| Hinzufügen von Live-Quellen zu Ihrem Quellstandort | 341 |
| Mit Kanälen arbeiten | 345 |
| Kanäle erstellen | 346 |
| Verwenden Sie Quellgruppen mit den Ausgängen Ihres Kanals | 349 |
| Kanäle löschen | 349 |
| Mit Programmen arbeiten | 350 |
| Programme erstellen | 350 |
| Mit Programmregeln arbeiten | 358 |
| Zielgruppen definieren | 358 |
| Fügen Sie Anzeigen und Werbeunterbrechungen ein | 361 |
| Anzeigeneinfügung einrichten | 362 |
| SCTE-35 Nachrichten für Werbeunterbrechungen | 363 |
| Zeitversetzte Anzeige | 370 |
| Zeitversetzende Parameter für Manifestanfragen | 371 |
| Verwendung von zeitversetzter Anzeige mit CDNs | 373 |
| Sicherheit | 376 |
| Datenschutz | 377 |
| Datenverschlüsselung | 378 |
| Identitäts- und Zugriffsverwaltung | 378 |

| | |
|--|-----|
| Zielgruppe | 379 |
| Authentifizierung mit Identitäten | 379 |
| Verwalten des Zugriffs mit Richtlinien | 383 |
| Wie AWS Elemental MediaTailor funktioniert mit IAM | 386 |
| Beispiele für identitätsbasierte Richtlinien | 393 |
| Beispiele für eine ressourcenbasierte Richtlinie | 396 |
| AWS verwaltete Richtlinien | 397 |
| Verwenden von serviceverknüpften Rollen | 399 |
| Fehlerbehebung | 402 |
| Compliance-Validierung | 405 |
| Ausfallsicherheit | 406 |
| Sicherheit der Infrastruktur | 406 |
| Serviceübergreifende Confused-Deputy-Prävention | 407 |
| Protokollierung und Überwachung | 409 |
| CloudWatch Alarme | 409 |
| CloudTrail Logs | 409 |
| AWS Trusted Advisor | 409 |
| Überwachung und Tagging | 411 |
| Berechtigungen für Amazon einrichten CloudWatch | 412 |
| Überwachung mithilfe von CloudWatch Protokollen | 413 |
| Steuerung des Volumens der Sitzungsprotokolle zur Anzeigenschaltung | 414 |
| Debug-Protokolle werden generiert | 416 |
| Als Run-Protokoll für Channel Assembly | 421 |
| Wird CloudWatch zum Anzeigen MediaTailor ADS von Protokollen verwendet | 426 |
| Überwachung mit CloudWatch Metriken | 466 |
| AWS Elemental MediaTailor CloudWatch Metriken | 467 |
| AWS Elemental MediaTailor CloudWatch Abmessungen | 475 |
| APIAnrufe protokollieren mit AWS CloudTrail | 476 |
| AWS Elemental MediaTailor Informationen in CloudTrail | 476 |
| Grundlagen zu AWS Elemental MediaTailor -Protokolldateieinträgen | 477 |
| Überwachung der Ressourcen zur Kanalmontage mitMediaTailorWarnungen | 479 |
| Benachrichtigungen anzeigen | 485 |
| Umgang mit Alarmen | 487 |
| Markieren von Ressourcen | 488 |
| Unterstützte Ressourcen | 488 |
| Tag (Markierung)-Einschränkungen | 489 |

| | |
|---|-------|
| Verwalten von Tags | 489 |
| Fehlerbehebung | 490 |
| Behebung von Wiedergabefehlern | 490 |
| Client-Fehler | 491 |
| Serverfehler | 493 |
| Beispiele | 496 |
| Kontingente | 497 |
| Kontingente für das Einfügen von Anzeigen | 497 |
| Kontingente für die Kanalmontage | 502 |
| MediaTailor Ressourcen | 507 |
| Dokumentverlauf | 509 |
| | dxxvi |

Was ist AWS Elemental MediaTailor?

AWS Elemental MediaTailor ist ein skalierbarer Service zum Einfügen von Anzeigen und zur Kanalmontage, der in der AWS Cloud ausgeführt wird. Mit MediaTailor können Sie Zuschauern gezielte Werbeinhalte bereitstellen und lineare Streams erstellen, während Sie gleichzeitig die Übertragungsqualität in over-the-top (OTT) Videoanwendungen beibehalten. MediaTailor Das Einfügen von Anzeigen 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 Systemen zur Anzeigenverfolgung: 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 -35 Markierungen versehen musst. SCTE 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 im gesamten AWS Elemental MediaTailor Benutzerhandbuch verwendet werden.

Konzepte zum Einfügen von Anzeigen

Hier finden Sie einen Überblick über die Konzepte im Zusammenhang mit der Anzeigenschaltung.

Entscheidungsserver anzeigen (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 zum Ursprungsserver und zum Anzeigenentscheidungsserver (ADS). Die Konfiguration enthält auch Endpunkte, die Eingangs- und Ausgangspunkte bereitstellen. MediaTailor

Dynamische Transcodierung

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 anhand der VAST Antwort des Anzeigenentscheidungservers (ADS) bestimmt. MediaTailor führt im Verlauf der Wiedergabe das Einfügen oder Ersetzen von Anzeigen in den Inhaltsstream durch.

VAST und VMAP

Vorlage für die Schaltung von VAST Videoanzeigen () und Playlist für mehrere VMAP Videoanzeigen () sind XML Antworten, die der Anzeigenentscheidungsserver auf Anzeigenanfragen von sendet MediaTailor. Die Antworten bestimmen, welche Anzeigen in das Manifest MediaTailor eingefügt werden. VMAP beinhaltet auch den Zeitplan für die Verfügbarkeit von Anzeigen. Weitere Informationen zur Logik hinter dem Einfügen von MediaTailor Anzeigen finden Sie unter [Das Verhalten bei der MediaTailor Anzeigeneinblendung verstehen](#). Weitere Informationen zur MediaTailor Funktionsweise von finden Sie unter [the section called “VAST, und Anforderungen für VMAP Werbeserver VPAID”](#). VAST

Konzepte für die Kanalmontage

Hier finden Sie einen Überblick über die Konzepte, die sich auf die Kanalmontage beziehen.

Kanäle

Ein Kanal fügt Ihre Quellmanifeste zu einem linearen Stream zusammen. Jeder Kanal hat einen oder mehrere Ausgänge, die die Wiedergabe ermöglichen, 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.

Konfiguration Package

Eine Packager-Konfiguration ist eine Darstellung Ihrer VOD Quelle, die bestimmte Merkmale des Paketformats enthält. Sie verknüpfen Ihre Paketkonfigurationen mit Kanalausgaben, 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.

Standorte der Quellen

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 handeln MediaPackage.

VODQuellen

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.

Publikum

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

MediaTailor interagiert zwischen Ihrem Content Delivery Network (CDN), dem Ursprungsserver und dem Ad Decision Server (ADS), um personalisierte Anzeigen in Live- und Video-on-Demand-Inhalte einzubinden.

Im Folgenden finden Sie eine Übersicht darüber, wie das Einfügen von MediaTailor Anzeigen funktioniert:

1. Ein Spieler oder CDN ähnliches wie Amazon CloudFront sendet eine Anfrage MediaTailor für HLS oder DASH Inhalt. Die Anfrage enthält Parameter des Players mit Informationen über den Zuschauer, die zur Personalisierung der Werbung verwendet werden.
2. MediaTailor ruft das Inhaltsmanifest und die Anzeigenspezifikationen ab, um die Anfrage zu bearbeiten:
 - MediaTailor sendet eine Anfrage an die ADS, die die Zuschauerinformationen enthält. Der ADS wählt Anzeigen auf der Grundlage der Zuschauerinformationen und der aktuellen Werbekampagnen aus. Es gibt die Anzeige URLs MediaTailor in einer VAST VMAP Oder-Antwort zurück.
 - MediaTailor manipuliert das Manifest so, dass es die Anzeigen enthält, die von der zurückgegeben wurden ADS, und transkodiert, damit sie den Kodierungsmerkmalen des ursprünglichen Inhalts entsprechen.

Wenn eine Anzeige noch nicht so transkodiert wurde, dass sie dem Inhalt entspricht, MediaTailor wird das Einfügen übersprungen und sie wird verwendet, MediaConvert um die Anzeige so vorzubereiten, dass sie für die nächste Anfrage bereit ist.
3. MediaTailor manipuliert das Manifest so, dass es das für die URLs Anzeigen enthält.
4. MediaTailor gibt das vollständig personalisierte Manifest an den anfragenden Spieler CDN oder Spieler zurück.

Die aufgerufenen Anzeigen werden anhand von Meilensteinen wie dem Beginn der Anzeige, der Mitte der Anzeige und dem Ende der Anzeige ADS erfasst. Im Verlauf der Wiedergabe MediaTailor sendet der Player oder der Player Werbe-Tracking-Beacons an das ADS Anzeigen-TrackingURL, 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.

Hinweise zu den ersten Schritten mit der Anzeigeneinfügung finden Sie unter. [Erste Schritte mit MediaTailor](#)

Zugehörige Services

- Amazon CloudFront ist ein globaler Service für das Content Delivery Network (CDN), der Daten und Videos sicher an Ihre Zuschauer liefert. Wird verwendet 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. Wird AWS Elemental MediaPackage als Ursprungsserver verwendet, um Inhalte vorzubereiten und Werbemarkierungen hinzuzufügen, bevor Streams an gesendet werden. MediaTailor Weitere Informationen zur MediaTailor Funktionsweise mit Originalservern finden Sie unter [Wie funktioniert das Einfügen von MediaTailor Anzeigen](#).
- AWS Identity and Access Management (IAM) ist ein Webdienst, mit dem Sie den Zugriff Ihrer Benutzer auf AWS Ressourcen sicher kontrollieren können. IAM steuert, wer Ihre AWS Ressourcen nutzen kann (Authentifizierung) und welche Ressourcen sie auf welche Weise verwenden 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 Nutzung der Dienste 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 Verträge oder Mindestverpflichtungen für die Nutzung MediaTailor. Ihnen werden Gebühren basierend auf Ihrer Nutzung des Service berechnet. Weitere Informationen finden Sie unter [MediaTailorPreise](#).

Regionen für MediaTailor

Um die Datenlatenz in Ihren Anwendungen zu reduzieren, MediaTailor bietet es regionale Endpunkte, an denen Sie Ihre Anfragen stellen können. 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 eine AWS-Konto](#)
- [Erstellen Sie einen Benutzer mit Administratorzugriff](#)

Melden Sie sich an für eine 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/signup>.
2. Folgen Sie den Online-Anweisungen.

Bei der Anmeldung müssen Sie auch einen Telefonanruf entgegennehmen und einen Verifizierungscode über die Telefontasten eingeben.

Wenn Sie sich für eine anmelden AWS-Konto, Root-Benutzer des AWS-Kontos wird eine erstellt. Der Root-Benutzer hat Zugriff auf alle AWS -Services und Ressourcen des Kontos. Aus Sicherheitsgründen sollten Sie einem Benutzer Administratorzugriff zuweisen und nur den Root-Benutzer verwenden, um [Aufgaben auszuführen, für die Root-Benutzerzugriff erforderlich](#) ist.

AWS sendet Ihnen nach Abschluss des Anmeldevorgangs eine Bestätigungs-E-Mail. Sie können jederzeit Ihre aktuelle Kontoaktivität anzeigen und Ihr Konto verwalten. Rufen Sie dazu <https://aws.amazon.com/> auf und klicken Sie auf Mein Konto.

Erstellen Sie einen Benutzer 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 Sie einen Benutzer 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 Benutzer in IAM Identity Center Administratorzugriff.

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

Melden Sie sich als Benutzer mit Administratorzugriff an

- 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 Zugriffsportal](#).

Weisen Sie weiteren Benutzern Zugriff zu

1. Erstellen Sie in IAM Identity Center einen Berechtigungssatz, der der bewährten Methode zur Anwendung von Berechtigungen mit den geringsten Rechten folgt.

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

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

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

Erste Schritte mit MediaTailor

Wählen Sie das entsprechende Thema aus, um mit dem Einfügen von MediaTailor Anzeigen oder der Erstellung von Kanälen zu beginnen.

Themen

- [Erste Schritte mit der MediaTailor Anzeigeneinfügung](#)
- [Erste Schritte mit der MediaTailor Kanalmontage](#)

Erste Schritte mit der MediaTailor Anzeigeneinfügung

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 Ihre DASH Inhaltsstreams vor
- Konfigurieren Sie eine Vorlage für einen Anzeigenentscheidungsserver (ADS) URL
- 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 Sie fertig sind, können Sie eine Wiedergabeanfrage MediaTailor für personalisierte Werbeinhalte in Ihrem Stream senden.

Themen

- [Voraussetzungen](#)
- [Schritt 1: Zugriff AWS Elemental MediaTailor](#)
- [Schritt 2: Bereiten Sie einen Stream vor](#)
- [Schritt 3: Konfigurieren Sie die ADS Anfrage URL - und Abfrageparameter](#)
- [Schritt 4: Erstellen einer Konfiguration](#)
- [Schritt 5: Testen Sie die Konfiguration](#)

- [Schritt 6: Senden Sie die Wiedergabeanforderung an AWS Elemental MediaTailor](#)
- [Schritt 7 \(optional\): Aktivität überwachen AWS Elemental MediaTailor](#)
- [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 Ursprungsserver so, dass er Manifeste für HLS oder DASH, die kompatibel mit sind, erstellt AWS Elemental MediaTailor.

Bereite einen HLS Stream vor

HLS Manifeste müssen die folgenden Anforderungen erfüllen:

- Manifeste müssen über das öffentliche Internet zugänglich sein.
- Manifeste müssen live oder als Video auf Abruf verfügbar sein (VOD).
- 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 optional für VOD Inhalte, für die 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` (üblicher) mit Dauern, wie im folgenden Beispiel gezeigt.

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

- `#EXT-X-DATERANGE` (seltener) mit Dauern wie im folgenden Beispiel gezeigt.

```
#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 MediaTailor Anzeigeneinblendung verstehen"](#).

- HLSMaster-Manifeste müssen der unter [HTTPLive-Streaming dokumentierten HLS Spezifikation entsprechen: Master-Playlist-Tags](#). Insbesondere muss #EXT-X-STREAM-INF die Felder RESOLUTION, BANDWIDTH und CODEC enthalten.

Nachdem Sie den Stream konfiguriert haben, notieren Sie sich das URL Quellpräfix für den Inhalt 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

DASHManifeste müssen die folgenden Anforderungen erfüllen:

- Manifeste müssen über das öffentliche Internet zugänglich sein.
- Manifeste müssen live oder als Video auf Abruf verfügbar sein (VOD).
- Manifeste müssen Ereignisse als Ad-Avails markieren und dazu entweder Splice-Insert-Marker oder Zeitsignal-Marker verwenden. Sie können die Anzeigenmarkierungen im Klartext XML oder in einer Base64-codierten Binärdatei angeben. Für das Einfügen von Spleißverbindungen muss der out-of-network Indikator aktiviert sein. Bei Zeitsignal-Markierungen muss es sich bei der ID des Segmentierungstyps, die sich innerhalb der Segmentierung befindet, um einen Cue-Out-Wert handelnUPID, 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, notieren Sie sich das URL Inhaltsursprungspräfix für das Manifest. DASH Sie benötigen es, um die Konfiguration später in AWS Elemental MediaTailor diesem Tutorial zu erstellen.

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

Um die erforderlichen Abfrageparameter zu ermitteln ADS, generieren Sie ein Anzeigen-Tag URL aus dem ADS. Dies URL dient als Vorlage für Anfragen an die ADS und besteht aus den folgenden Komponenten:

- Statische Werte
- Werte, die von AWS Elemental MediaTailor (bezeichnet durch `session` oder `avail` Abfrageparameter) generiert wurden
- Von Playern generierte Werte, die von der Client-Anwendung abgerufen wurden (gekennzeichnet durch `player_params` -Abfrageparameter)

Example Tag URL von einem hinzufügen 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 für jede Spielsitzung und führt zu einem anderen Wert für die Anfrage URL für jede Sitzung. VAST
- cust_params sind vom Player bereitgestellte dynamische Werte

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

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

Testen

Sie können eine statische VAST Antwort von Ihrem zu ADS Testzwecken verwenden. Im Idealfall gibt die VAST Antwort eine MP4 Wiedergabe in Mezzanine-Qualität zurück, die transkodiert werden kann. AWS Elemental MediaTailor Wenn die Antwort von mehrere Wiedergabeversionen ADS enthält, MediaTailor wählt die Wiedergabeversion mit der höchsten Qualität und Auflösung MP4 aus und sendet sie an den Transcoder.


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 die primäre Kennung für die Konfiguration. Die maximale zulässige Länge beträgt 512 Zeichen.
4. Geben Sie unter Videoinhaltsquelle das URL Präfix für das HLS Master-Manifest oder das DASH Manifest für diesen Stream ein, abzüglich der Asset-ID. Wenn das Master-Manifest beispielsweise URL lautet `http://origin-server.com/a/master.m3u8`, würden Sie Folgendes eingeben `http://origin-server.com/a/`. 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 Ihr Content Origin verwendet HTTPS, muss das zugehörige Zertifikat von einer bekannten Zertifizierungsstelle stammen. (Es darf kein selbstsigniertes Zertifikat sein.) Andernfalls kann AWS Elemental MediaTailor keine Verbindung zum Inhaltsursprung hergestellt werden und es können keine Manifeste als Antwort auf Spieleranfragen bereitgestellt werden.

5. Geben Sie für Ad Decision Server den URL für Ihren ein ADS. Dabei handelt es sich entweder um die Variablen URL with, wie unter beschrieben [Schritt 3: Konfigurieren Sie die ADS Anfrage URL - und Abfrageparameter](#), oder um VAST URL die statische Variablen, die Sie zu Testzwecken verwenden. Die maximale Länge beträgt 25.000 Zeichen.

 Note

Wenn Sie ADS das Zertifikat verwenden HTTPS, muss es von einer bekannten Zertifizierungsstelle stammen. (Es darf kein selbstsigniertes Zertifikat sein.) Das Gleiche gilt für die von ihnen URLs zurückgegebenen Anzeigen im Zwischengeschoss. ADS Andernfalls MediaTailor schlägt es fehl, Anzeigen vom Ursprung des Inhalts abzurufen und in die Manifeste einzufügen.

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

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

7. (Optional) Wenn Ihr Ursprungsserver DASH Manifeste mit einer Periode erstellt, wählen Sie den Herkunftstyp „DASHmpd manifest“ und dann „_“. SINGLE PERIOD 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 mit einem 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://bdaaeb4bd9114c088964e4063f849065.mediatailor.us-east-1.amazonaws.com
```

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

Beispiel

```
AKIAIOSFODNN7EXAMPLE
```

- `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://bdaaeb4bd9114c088964e4063f849065.mediatailor.us-east-1.amazonaws.com/v1/  
master/AKIAIOSFODNN7EXAMPLE/myOrigin/master.m3u8
```

- Beispiel: DASH

```
https://bdaaeb4bd9114c088964e4063f849065.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 gezeigt, den 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 CDN senden Sie Wiedergabeanfragen an den von der Konfiguration bereitgestellten Wiedergabeendpunkt AWS Elemental MediaTailor. Alle vom Spieler definierten dynamischen Variablen, die Sie in der ADS Anfrage verwendet haben, [Schritt 3: Konfigurieren Sie die ADS Anfrage URL - und Abfrageparameter](#) müssen URL in der Manifestanforderung des Players definiert werden.

Example

Gehen Sie davon aus, dass Ihre Vorlage die ADS URL folgende ist.

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

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 anstelle von `ADS`

Bei der Spieleraanfrage URL handelt es sich um eine Variante der folgenden HLS und DASH Beispiele.

```
https://bdaaeb4bd9114c088964e4063f849065.mediatailor.us-east-1.amazonaws.com/v1/master/
AKIAIOSFODNN7EXAMPLE/myOrigin/master.m3u8?ads.cust_params=viewerinfo
```

```
https://bdaaeb4bd9114c088964e4063f849065.mediatailor.us-east-1.amazonaws.com/v1/dash/
AKIAIOSFODNN7EXAMPLE/myOrigin/manifest.mpd?ads.cust_params=viewerinfo
```

Wenn AWS Elemental MediaTailor die Spieleraanfrage eingeht, werden die Spielervariablen auf der Grundlage der Informationen in der Anfrage definiert. Die daraus resultierende ADS Anfrage URL ist eine Variante davon.

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

Weitere Hinweise zur Konfiguration von Schlüssel-Wert-Paaren, die an die übergeben werden ADS, finden Sie unter [Verwendung dynamischer Anzeigenvariablen in AWS Elemental MediaTailor](#)

Schritt 7 (optional): Aktivität überwachen AWS Elemental MediaTailor

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 Diensten zu ermöglichen.

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

1. Öffnen Sie die IAM Konsole unter <https://console.aws.amazon.com/iam/>.
2. Wählen Sie im Navigationsbereich der IAM Konsole Rollen und anschließend Rolle erstellen aus.
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 Bestätigungsdiaologfeld 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, das Playback URL für deinen Kanal aufrufen 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 MediaTailor , um linear 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 Quellort 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 sowohl für VOD Quellinhalte als auch für Ad Slate über Ressourcen verfügen. Sie müssen den Pfad zu den Manifesten für die Assets kennen.

Note

Wenn Sie die automatische Kodierung mit adaptiver Bitrate (ABR) 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. AWS Elemental MediaPackage B.

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>. Das URL muss in einem HTTP URL Standardformat mit dem Präfix <http://> oder <https://> sein.
5. Wählen Sie Quellspeicherort erstellen.

Schritt 2: Fügen Sie VOD Quellen zu Ihrem Quellort hinzu

Nachdem du einen oder mehrere Quellorte für deinen Kanal definiert hast, kannst du eine oder mehrere VODQuellen 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 auch HLS als Paket enthalten ist DASH, können Sie für jedes Format zwei Paketkonfigurationen erstellen. Sie können dann die Quellgruppen der Paketkonfiguration verwenden, um zwei Kanalausgaben 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 VODQuelle hinzufügen aus.
5. Geben Sie unter VODQuelldetails einen Namen für Ihre VOD Quelle ein, z. my-example-videoB.

6. Unter Paketkonfigurationen > *source-group-name* geben Sie 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 [Verwenden Sie Quellgruppen mit den Ausgängen Ihres Kanals](#).
 - Typ: Wählen Sie das Paketformat für diese Konfiguration aus. MediaTailorunterstützt HLS undDASH.
 - Relativer Pfad: Der relative Pfad von der Base des Quellspeicherorts HTTP URL 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 Quelle für Ihre Anzeigenliste hinzuzufügen. VOD

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.
6. Definieren Sie unter Ausgabedetails die Einstellungen für diese Ausgabe:
 - Manifestname: Geben Sie einen Manifestnamen ein, z. B. **index**. 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 der Sie es erstellt haben [Schritt 1: Erstellen Sie einen Quellspeicherort](#).
 - 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.
 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.
 11. Überprüfen Sie Ihre Einstellungen im Bereich Überprüfen und erstellen.

12. Wählen Sie Create channel (Channel erstellen).

Note

Kanäle werden im gestoppten Zustand 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 Senderplan 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 Server für die Anzeigeneinblendung verwenden, z. B. 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.
 - VODQuellname: Wählen Sie „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 Namen des Programms 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 Slate-Quellspeicherorts:** Wählen Sie „Bestehenden Quellspeicherort auswählen“ und wählen Sie den Quellspeicherort aus, an dem Ihr Slate gespeichert ist, den Sie zuvor in diesem Tutorial erstellt haben.
- **VODQuellname:** Wählen Sie „Eine vorhandene VOD Quelle auswählen“ und wählen Sie die VOD Quelle aus, die Sie für Slate verwenden, 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 Programmquelle (alle Audio-, Video- und Untertitelspuren) ausgerichtet ist. Andernfalls wird die Werbeunterbrechung übersprungen. Wenn Sie beispielsweise 0 eingeben, wird eine Pre-Roll-Werbeunterbrechung erstellt, 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 du möchtest, kannst du MediaTailor damit personalisierte Werbung in die Werbeunterbrechungen deiner Programme im Stream des Kanals einfügen.

Voraussetzungen

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

- Sie benötigen einen Anzeigenentscheidungsserver (ADS).
- Sie müssen im Rahmen des [Mit Programmen arbeiten](#) 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: Geben Sie die Wiedergabe URL vom Ausgang Ihres Kanals ein, abzüglich des Dateinamens und der Erweiterung. Weiterführende Informationen zur MediaTailor Konfiguration finden Sie unter [Erforderliche Einstellungen](#).
 - Entscheidungsserver hinzufügen: Geben Sie den URL für Ihren einADS.
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 [Konfiguration MediaTailor als Ihr Dienst zur Anzeigeneinfügung](#).

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 HTTP 4xx 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 Ihr Kanal ordnungsgemäß funktioniert, öffnen Sie einen Webbrowser und geben Sie den URL aus der Ausgabe Ihres 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. Um deinen Kanal für später zu behalten, 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.

Konfiguration MediaTailor als Ihr Dienst zur Anzeigeneinfügung

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

Wenn Sie ein Content Distribution Network (CDN) mit verwenden MediaTailor, müssen Sie die Verhaltensregeln in der einrichten, CDN bevor Sie CDN Informationen zur Konfiguration hinzufügen. Weitere Informationen zur Einrichtung Ihres CDN finden Sie unter [Integration eines CDN](#).

Themen

- [Unterstützte Audio- und Videocodecs](#)
- [VAST, und Anforderungen für VMAP Werbeserver VPAID](#)
- [Mit MediaTailor Konfigurationen arbeiten](#)
- [Anpassen des Verhaltens von Werbeunterbrechungen](#)
- [Integration einer Inhaltsquelle](#)
- [Overlay-Anzeigen](#)
- [Werbe-ID-Dekoration](#)
- [Daten zur Berichterstattung und Nachverfolgung](#)
- [Verwendung dynamischer Anzeigenvariablen in AWS Elemental MediaTailor](#)
- [Arbeiten mit CDNs](#)
- [Das Verhalten bei der MediaTailor Anzeigeneinblendung verstehen](#)
- [Grundlegendes zur MediaTailor servergesteuerten Anzeigeneinfügung](#)

Unterstützte Audio- und Videocodecs

MediaTailor unterstützt die folgenden Codecs.

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

VAST, und Anforderungen für VMAP Werbeserver VPAID

Um Ihren Anzeigenserver mit zu integrieren AWS Elemental MediaTailor, muss Ihr Anzeigenserver eine Nachricht senden XML, die den IAB Spezifikationen für die unterstützten Versionen von VAST und VMAP entspricht. Sie können einen öffentlichen VAST Validator verwenden, um sicherzustellen, dass Ihre Tags korrekt formatiert sind.

AWS Elemental MediaTailor Unterstützungen VAST und VMAP Antworten von Anzeigenentscheidungsservern. AWS Elemental MediaTailor unterstützt auch die Weiterleitung von VPAID Metadaten über unser clientseitiges Reporting API zur kundenseitigen Anzeigeneinblendung. Weitere Informationen zur clientseitigen Berichterstellung finden Sie unter [Kundenseitiges Tracking](#).

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

- [VAST2.0 und 3.0](#)
- [VMAP1.0](#)
- [VPAID2,0](#)

VASTAnforderungen

Die VAST Antwort Ihres Anzeigenservers muss IAB konforme TrackingEvents Elemente und Standardereignistypen enthalten, wie `impression` z. Wenn du keine standardmäßigen Tracking-Ereignisse einfügst, AWS Elemental MediaTailor lehnt die VAST Antwort ab und stellt keine nutzlose Anzeige bereit.

VASTIn Version 3.0 wurde die Unterstützung für Ad-Pods eingeführt, d. h. die Bereitstellung einer Reihe sequentieller linearer Anzeigen. Wenn eine bestimmte Anzeige in einem Werbe-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 AWS Elemental MediaTailor](#).

AWS Elemental MediaTailor leitet beim Senden der VAST Anzeigenserver-Anfrage `user-agent` und bei serverseitigen Tracking-Aufrufen die `x-forwarded-for` Header des Players als Proxy weiter. Stellen Sie sicher, dass Ihre Werbeserver diese Header verarbeiten können. Alternativ können Sie `[session.user_agent]` oder verwenden `[session.client_ip]` und diese Werte in Abfragezeichenfolgen für das Anzeigen-Tag und die Anzeige übergeben. URL Weitere Informationen finden Sie unter [Verwenden von Sitzungsvariablen](#).

Anzeigenanrufe

AWS Elemental MediaTailor ruft Ihre VAST Anzeigen URL wie in Ihrer Konfiguration definiert auf. Es ersetzt beim Anzeigenaufruf alle spieler- oder sitzungsspezifischen Parameter. MediaTailor folgt bis zu fünf Ebenen von VAST Wrappern und Weiterleitungen in der Antwort. VAST Führt in Live-Streaming-Szenarien zu Beginn der Werbeanzeige für verbundene Spieler gleichzeitig Anzeigenrufe 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.

Indizierung von Werbemitteln

AWS Elemental MediaTailor Indiziert jedes Creative eindeutig anhand des Werts des im Element angegebenen Attributs. `id <Creative>` Wenn die ID eines Kreativen nicht angegeben ist, wird die Mediendatei URL 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, wird der gespeicherte Inhalt MediaTailor verwendet, anstatt den eingehenden Inhalt erneut zu verarbeiten. Wenn du eine Creative-ID MediaTailor wiederverwendest, die ältere, gespeicherte Anzeige verwendest und deine neue Anzeige nicht abspielt.

VPAID Anforderungen

VPAID ermöglicht es Publishern, hochgradig interaktive Videoanzeigen zu schalten und Kennzahlen zur Sichtbarkeit ihrer monetarisierten Streams bereitzustellen. [Informationen zu finden Sie in VPAID der Spezifikation. VPAID](#)

AWS Elemental MediaTailor unterstützt eine Mischung aus server-side-stitched VAST MP4 linearen Anzeigen und client-side-inserted VPAID interaktiven Werbemitteln in derselben Anzeigenauswahl. Dabei wird die Reihenfolge beibehalten, in der sie in der Antwort erscheinen. VAST MediaTailor folgt VPAID Weiterleitungen durch maximal fünf Ebenen von Wrappern. Die clientseitige Berichtsantwort enthält die unverpackten Metadaten. VPAID

Beachten Sie bei der Verwendung VPAID die folgenden Richtlinien:

- Konfiguriere einen MP4 Slate für deine VPAID Kreativen. AWS Elemental MediaTailor füllt die VPAID Anzeigenfelder mit Ihrem konfigurierten Slate und stellt VPAID Anzeigen-Metadaten bereit, die der Kundenplayer zur Schaltung der interaktiven Anzeigen verwenden kann. Wenn Sie kein Slate konfiguriert haben, werden die VPAID Anzeigenmetadaten bei der MediaTailor Schaltung einer Anzeige wie gewohnt über clientseitige Berichte bereitgestellt. Außerdem wird eine Fehlermeldung CloudWatch bezüglich des fehlenden Slate protokolliert. Weitere Informationen erhalten Sie unter [Schiefer wird eingefügt](#) und [Konfiguration erstellen](#).
- Verwenden Sie die clientseitige Berichterstattung. AWS Elemental MediaTailor unterstützt VPAID durch unser kundenseitiges Reporting. API Weitere Informationen finden Sie unter [Kundenseitiges Tracking](#).

Es ist theoretisch möglich, den standardmäßigen serverseitigen Berichtsmodus mit zu verwenden. VPAID Wenn Sie jedoch serverseitige Berichte verwenden, gehen alle Informationen über das Vorhandensein der VPAID Anzeige und die sie umgebenden Metadaten verloren, da diese nur auf der Clientseite verfügbar sind. API

- Stellen Sie in Live-Szenarien sicher, dass Ihre Anzeigenverkäufe, gekennzeichnet durch, groß genug sind `EXT-X-CUE-OUT: Duration`, um jegliche Nutzerinteraktivität zu ermöglichen. VPAID Wenn in beispielsweise eine VPAID Anzeige VAST XML angegeben ist, die 30 Sekunden lang ist, müssen Sie festlegen, dass Ihre Anzeigenverfügbarkeit mehr als 30 Sekunden beträgt, um die Anzeige aufzunehmen. Wenn Sie dies nicht tun, gehen die VPAID Metadaten verloren, da die verbleibende Dauer der Anzeigenverfügbarkeit nicht lang genug ist, um die VPAID Anzeige aufzunehmen.

Mit MediaTailor Konfigurationen arbeiten

In diesem Abschnitt wird beschrieben, wie Sie Konfigurationen erstellen, anzeigen, bearbeiten und löschen.

Themen

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

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

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, dem REST API AWS CLI, 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 einen CDN mit AWS Elemental MediaTailor für Manifest- und Berichtsanfragen ein. Sie können die Konfigurationswiedergabe URLs für das CDN Setup verwenden. Hinweise zum Einrichten von Manifest CDN - und Berichtsanfragen finden Sie unter [Integration eines CDN](#).

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 abzüglich der Asset-ID ein. Die maximale Länge beträgt 512 Zeichen.

Das URL Präfix `http://origin-server.com/a/` ist beispielsweise gültig für ein HLS übergeordnetes Manifest URL von `http://origin-server.com/a/main.m3u8` und für ein DASH Manifest URL 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 Ihr Content Origin verwendet HTTPS, muss das zugehörige 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 den URL für Ihren Anzeigenentscheidungsserver ein (ADS). Dabei handelt es sich entweder um die Variablen URL with, wie unter beschrieben [Schritt 3: Konfigurieren Sie die ADS Anfrage URL - und Abfrageparameter](#), oder um VAST URL die statische Variablen, die Sie zu Testzwecken verwenden. Die maximale Länge beträgt 25.000 Zeichen.

Note

Wenn Sie ADS das Zertifikat verwenden HTTPS, muss es von einer bekannten Zertifizierungsstelle stammen. Es kann kein selbstsigniertes Zertifikat sein. Das Gleiche gilt auch für die von ihnen zurückgesandte Mezzanine-AnzeigeURLs. ADS 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 oder () konfigurieren. MediaTailor API AWS Command Line Interface AWS CLI

Konfigurations-Aliase

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

Parametervariable des Players

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](#).

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](#).

Einzelheiten zur Personalisierung

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

Slate-Werbung

Geben Sie den Wert URL für ein qualitativ hochwertiges MP4 Asset ein, das transkodiert und zum Ausfüllen von Zeit verwendet werden soll, die nicht für Werbeanzeigen verwendet wird. AWS Elemental MediaTailor zeigt die Tafel an, um Lücken in Medieninhalten zu füllen. Die Konfiguration des Slate ist für VPAID Nichtkonfigurationen optional. Dafür 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 Ihr Slate 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

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

Endstoßstange

Der Standort URL der Anlage am Ende der Stoßstange. Bumper sind kurze Video- oder Audioclips, die zu Beginn oder am Ende einer Werbeunterbrechung abgespielt werden. Sie können auf Amazon S3 oder einem anderen Speicherdienst gespeichert werden. Weitere Informationen über Stoßstangen finden Sie unter [Stoßstangen einfügen](#).

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 bezieht sich eher auf das Ersetzen von Anzeigen in Live- und VOD Streams als auf das Einfügen von Anzeigen, da sie auf einem zugrundeliegenden Inhaltsstream basiert. Weitere Informationen zum Verhalten von Werbeunterbrechungen, einschließlich des Ersetzens und Einfügens von Anzeigen, finden Sie unter [Das Verhalten bei der MediaTailor Anzeigeneinblendung verstehen](#).

Live-Pre-Roll-Werbeentscheidungsserver

Um Anzeigen zu Beginn eines Livestreams einzufügen, bevor der Hauptinhalt wiedergegeben wird, geben Sie den Pre-Roll URL für die Anzeige vom Anzeigenentscheidungsserver ein (ADS). Dabei handelt es sich entweder um die Variablen URL with, wie unter beschrieben [Schritt 3: Konfigurieren Sie die ADS Anfrage URL - und Abfrageparameter](#), oder um VAST URL die statische Variablen, die Sie zu Testzwecken verwenden. Die maximale Länge beträgt 25.000 Zeichen.

Note

Wenn Sie ADS das Zertifikat verwenden HTTPS, muss es von einer bekannten Zertifizierungsstelle stammen. Es kann kein selbstsigniertes Zertifikat sein. Das Gleiche gilt auch für die von ihnen zurückgesandte Mezzanine-AnzeigeURLs. ADS 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 Anzeigen zu Beginn eines Livestreams einfügen, geben Sie die maximal zulässige Dauer für die Verfügbarkeit der Pre-Roll-Anzeige ein. MediaTailor wird diese Dauer beim Einfügen von Anzeigen nicht überschreiten. Wenn die Antwort von mehr Anzeigen ADS 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 dem AWS Command Line Interface (AWS CLI) oder mit dem konfigurieren MediaTailor API.

CDNPräfix für Inhaltssegmente

Ermöglicht AWS Elemental MediaTailor das Erstellen von Manifesten mit URLs Ihrem CDN Pfad für Inhaltssegmente. Bevor Sie diesen Schritt ausführen, richten Sie eine Regel in Ihrem ein, CDN nach der Segmente von Ihrem Ursprungsserver abgerufen werden sollen. Geben Sie als Präfix für das CDN Inhaltssegment den CDN Präfixpfad ein.

Weitere Informationen zur Integration MediaTailor mit einem CDN finden Sie unter [Arbeiten mit CDNs](#).

CDNPräfix für Anzeigensegmente

Ermöglicht AWS Elemental MediaTailor das Erstellen 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 für das Präfix für das CDN Anzeigensegment ausfüllen können, müssen Sie in Ihrem eine Regel einrichten, CDN nach der Anzeigensegmente von der folgenden Herkunft abgerufen werden sollen, wie im folgenden Beispiel.

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

Geben Sie für das Präfix für das CDN Anzeigensegment den Namen Ihres CDN Präfixes in der Konfiguration ein.

Weitere Informationen zur Integration MediaTailor mit einem CDN finden Sie unter [Arbeiten mit CDNs](#).

DASHTyp des Ursprungs-Manifests

Wenn Ihr Ursprungsserver DASH Manifeste für einzelne Perioden erzeugt, öffnen Sie die Drop-down-Liste und wählen Sie SINGLE _ . PERIOD MediaTailor. Manifeste werden standardmäßig als Manifeste mit mehreren Perioden behandelt. Weitere Informationen finden Sie unter [the section called “Integration einer MPEG DASH -Quelle”](#).

DASHMPD-Standort

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

- Sie richten CDN Routing-Regeln für den Zugriff auf MediaTailor Manifeste ein.
- Sie verwenden clientseitige Berichterstattung, oder Ihr Player unterstützt HTTP Sticky-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 mit Hilfe des AWS Support bereits benutzerdefinierte Profile eingerichtet haben.

Passthrough für Werbemarkierungen

Für HLS, aktiviert oder deaktiviert den Passthrough für Werbemarkierungen. Wenn das Passthrough für Werbemarkierungen aktiviert ist, werden MediaTailor EXT-X-CUE-IN, EXT-X-CUE-OUT, und EXT-X-SPLICEPOINT-SCDTE35 Anzeigenmarkierungen vom ursprünglichen Manifest zum personalisierten Manifest weitergeleitet. MediaTailor wendet keine Logik auf die Werte der Werbemarkierung an. 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 ist, wird der Wert 0 im personalisierten Manifest nicht geändert.

Konfiguration anzeigen

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.

AWS Elemental 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.

Eine 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 [Konfiguration erstellen](#).
4. Wählen Sie Save (Speichern) 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 Bestätigungsdialogfeld Delete (Löschen) den Text **Delete** ein und wählen Sie dann Delete (Löschen) aus.

Anpassen des Verhaltens von Werbeunterbrechungen

Wenn Sie eine Konfiguration erstellen, können Sie optionale Konfigurationseinstellungen für Werbeunterbrechungen angeben, die das Verhalten von Werbeunterbrechungen steuern. Wählen Sie das entsprechende Thema aus, um Informationen zu den folgenden Optionen zur Anpassung von Werbeunterbrechungen zu erhalten.

Themen

- [Die Unterdrückung von Werbeunterbrechungen konfigurieren](#)

Die Unterdrückung von Werbeunterbrechungen konfigurieren

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 Werbepplatzunterdrückung bezeichnet. In diesem Thema erfahren Sie, wie das geht, und es wird auch erklärt, wie die Konfiguration der Anzeigenunterdrückung funktioniert.

Die Anzeigenunterdrückung 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.

Anzeigenunterdrückung konfigurieren

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 Werbeunterdrückung ein- oder ausschalten 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.

- **AFTER_LIVE_EDGE**: personalisiert MediaTailor keine Werbeunterbrechungen, die innerhalb des Live-Edge-Bereichs liegen, zuzüglich des Werts für die Avail-Unterdrückung.
- **Avail suppression value** (Werbeplatz-Unterdrückungswert) – Ein Zeitpunkt relativ zur Live-Grenze in einem Livestream. Akzeptierter Wert: Ein Zeitwert in HH:MM:SS.
- Richtlinie „Avail-Unterdrückung auffüllen“ — Definiert die Richtlinie, die für den Avail-Unterdrückungsmodus 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. In diesem Abschnitt finden Sie Beispiele, die Ihnen einige dieser Interaktionen zeigen. Verwenden Sie diese Beispiele, um die Konfigurationsparameter für Ihre spezielle Situation einzurichten.

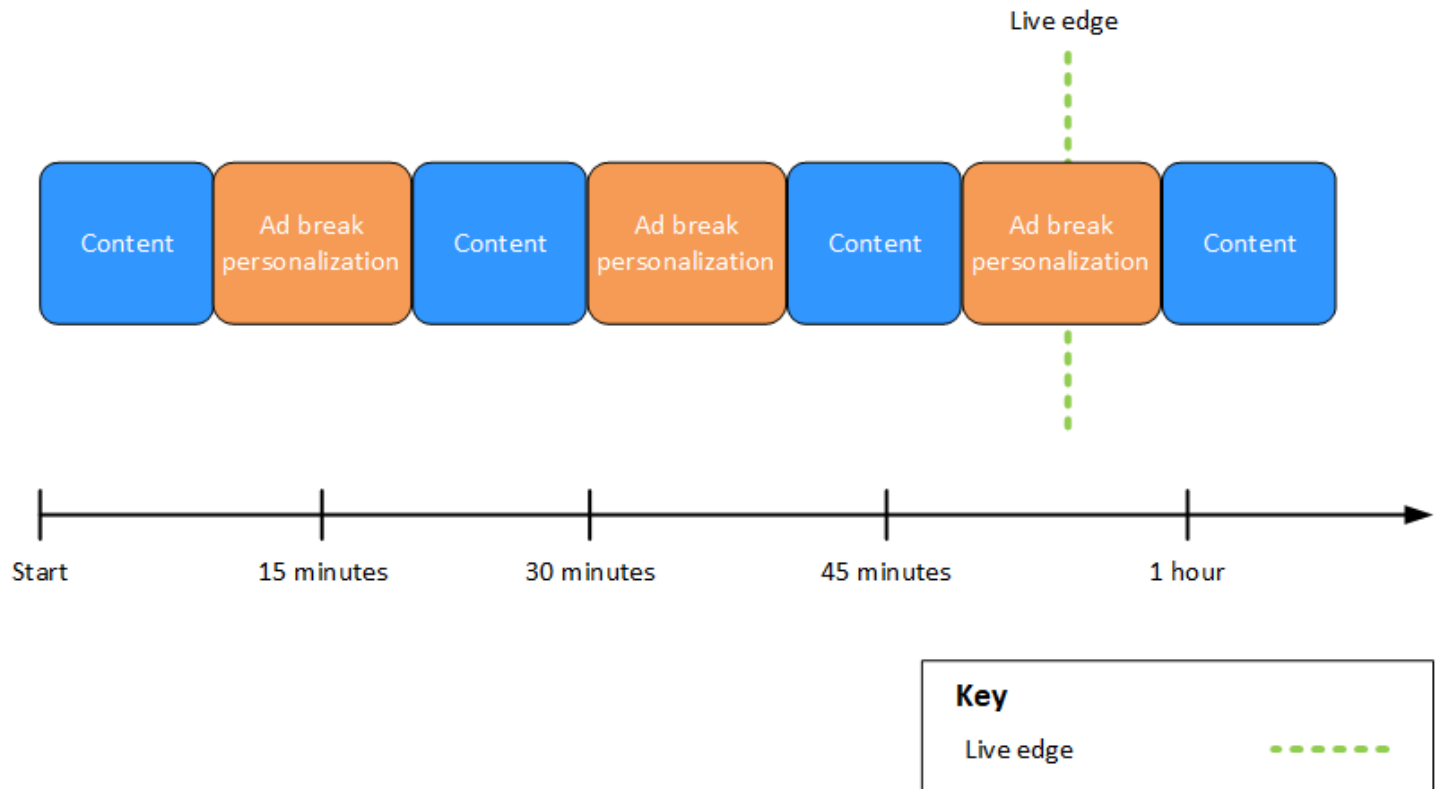
Im Folgenden finden Sie Beispiele für Einstellungen zur Anzeigenunterdrückung:

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

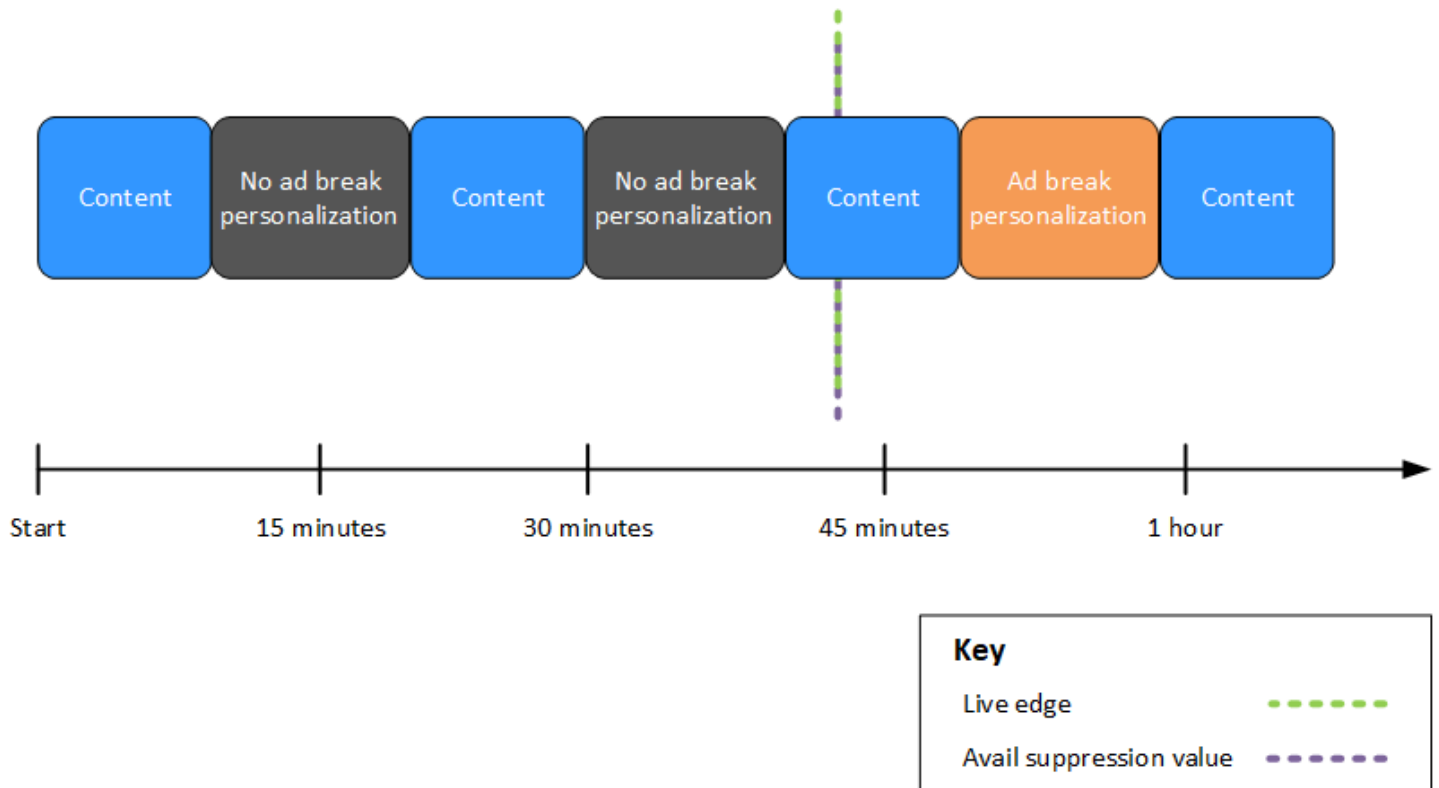
Wenn der Verfügungsunterdrückungsmodus auf **BEHIND_LIVE_EDGE** und der Wert für die Verfügbarkeitsunterdrückung auf eingestellt ist `00:00:00`, ist der Verfügbarkeitsunterdrückungswert mit dem Live-Edge synchronisiert. MediaTailor personalisiert keine Werbeunterbrechungen, die am oder vor dem Live-Edge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

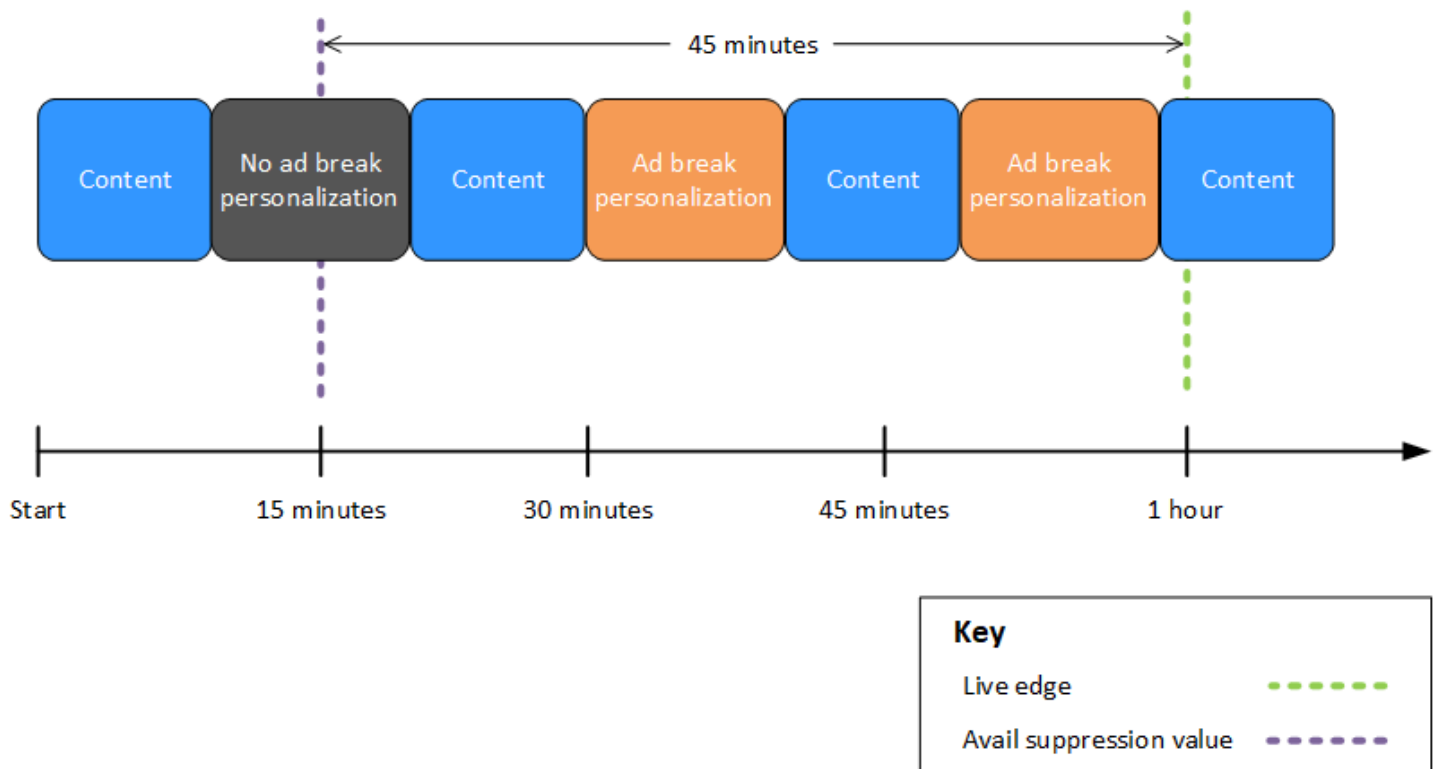
MediaTailor personalisiert in diesem Beispiel Werbeunterbrechungen, die innerhalb von 45 Minuten hinter dem Live-Edge beginnen. MediaTailor personalisiert keine Werbeunterbrechungen, die am oder nach 45 Minuten hinter dem Live-Edge beginnen.

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

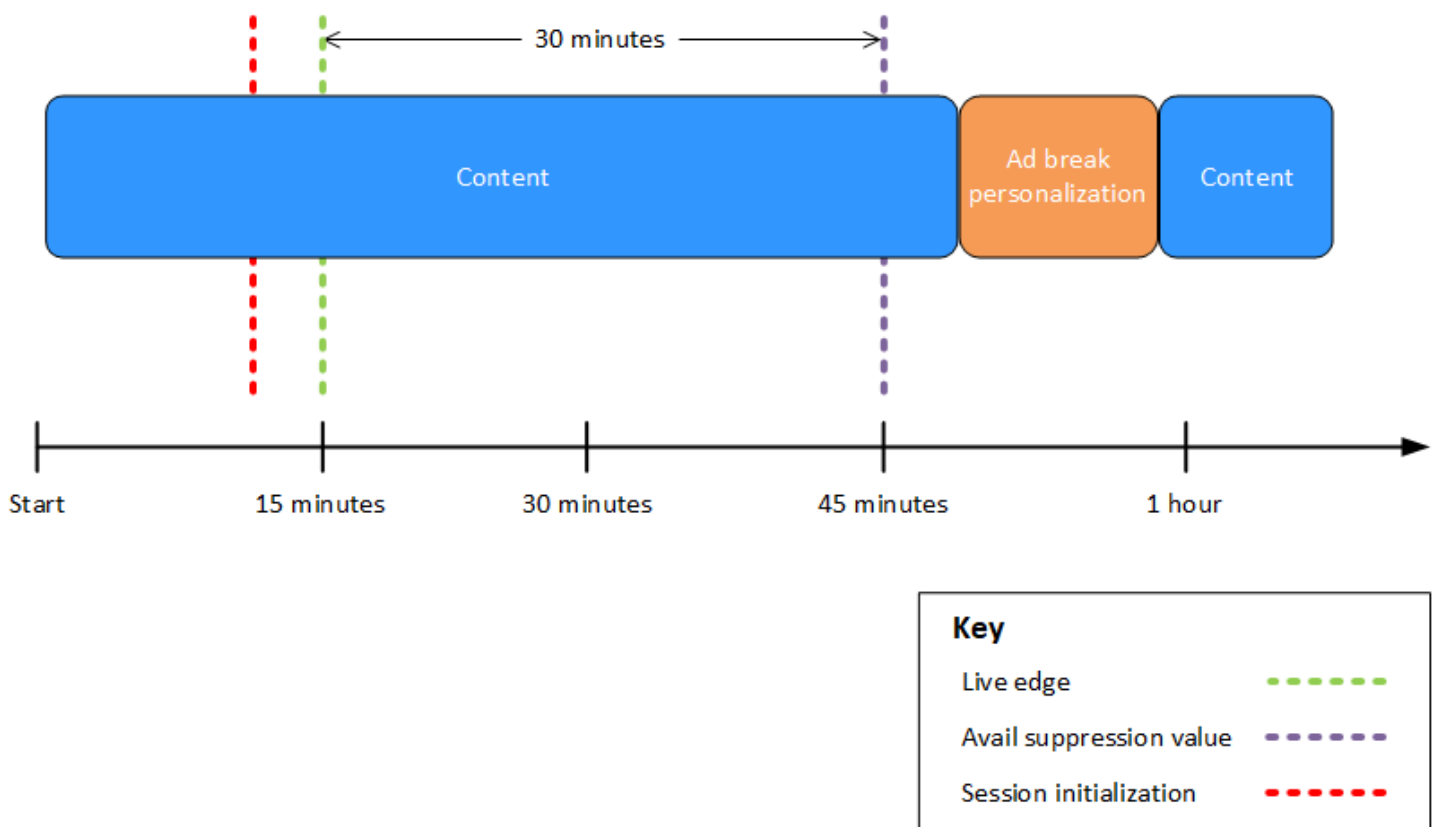
Wenn der Verfügbarkeitsunterdrückungsmodus auf eingestellt ist **AFTER_LIVE_EDGE** und der Wert für die Verfügbarkeitsunterbindung größer als Null ist, werden Werbeunterbrechungen erst personalisiert, wenn die verstrichene Zeit der Sitzung diesen Wert erreicht MediaTailor hat.

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_AVAILAusfüllen“ und einer aktiven Werbeunterbrechung am Ende des Zeitraums

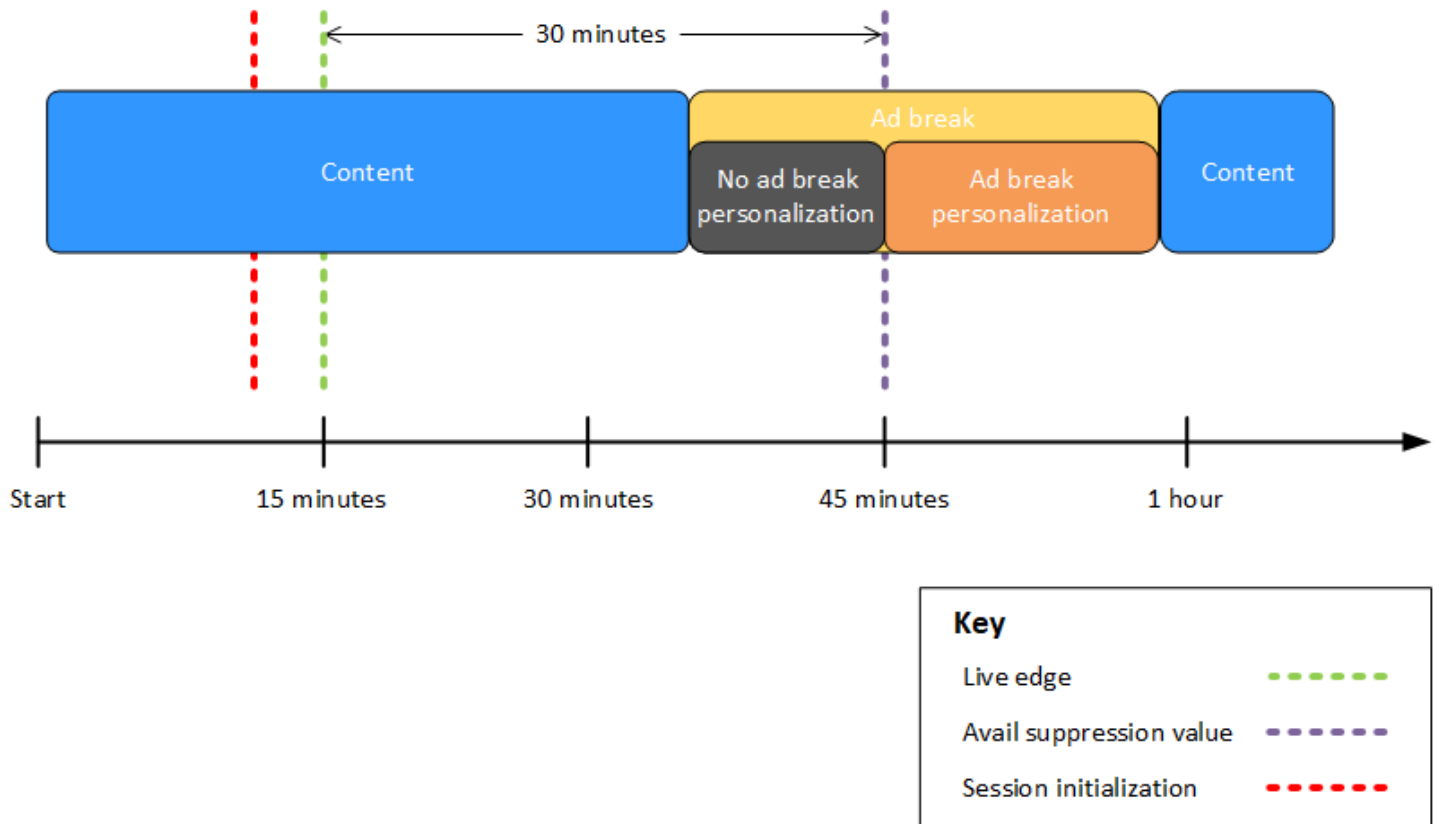
Wenn der Modus zur Verfügungsunterbindung auf eingestellt ist AFTER_LIVE_EDGE und der Wert für die Verfügbarkeitsunterbindung größer als Null ist, werden Werbeunterbrechungen erst personalisiert, wenn die verstrichene Zeit der Sitzung diesen Wert erreicht MediaTailor hat.

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-EdgePARTIAL_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.

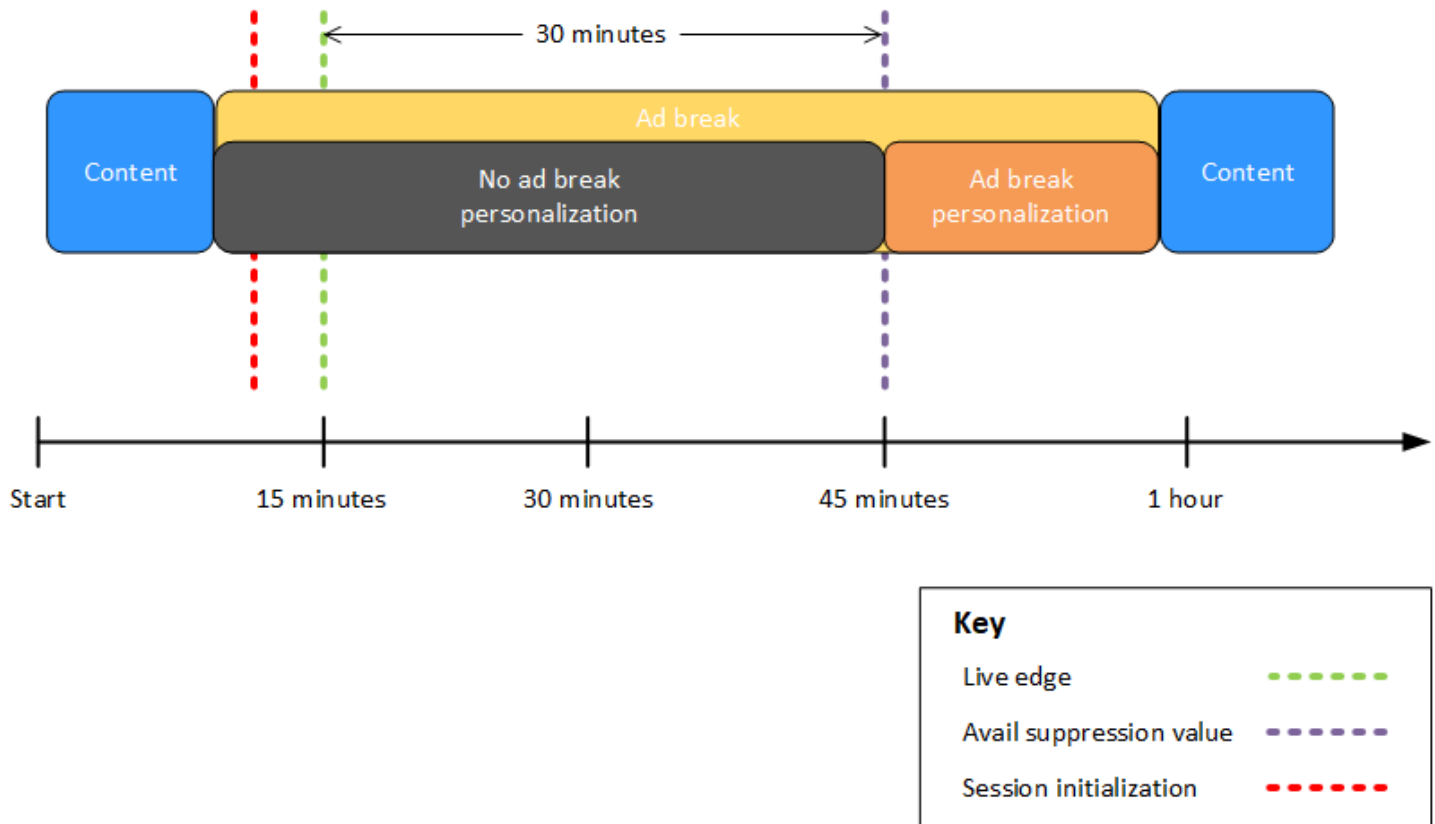
In der folgenden Abbildung sind verschiedene Blöcke horizontal entlang einer Zeitleiste angeordnet, die von links nach rechts verläuft. Jeder Block steht für einen Zeitraum, in dem der Inhalt eines Live-Streams, einer personalisierten Werbepause oder einer nicht personalisierten Werbepause abgespielt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

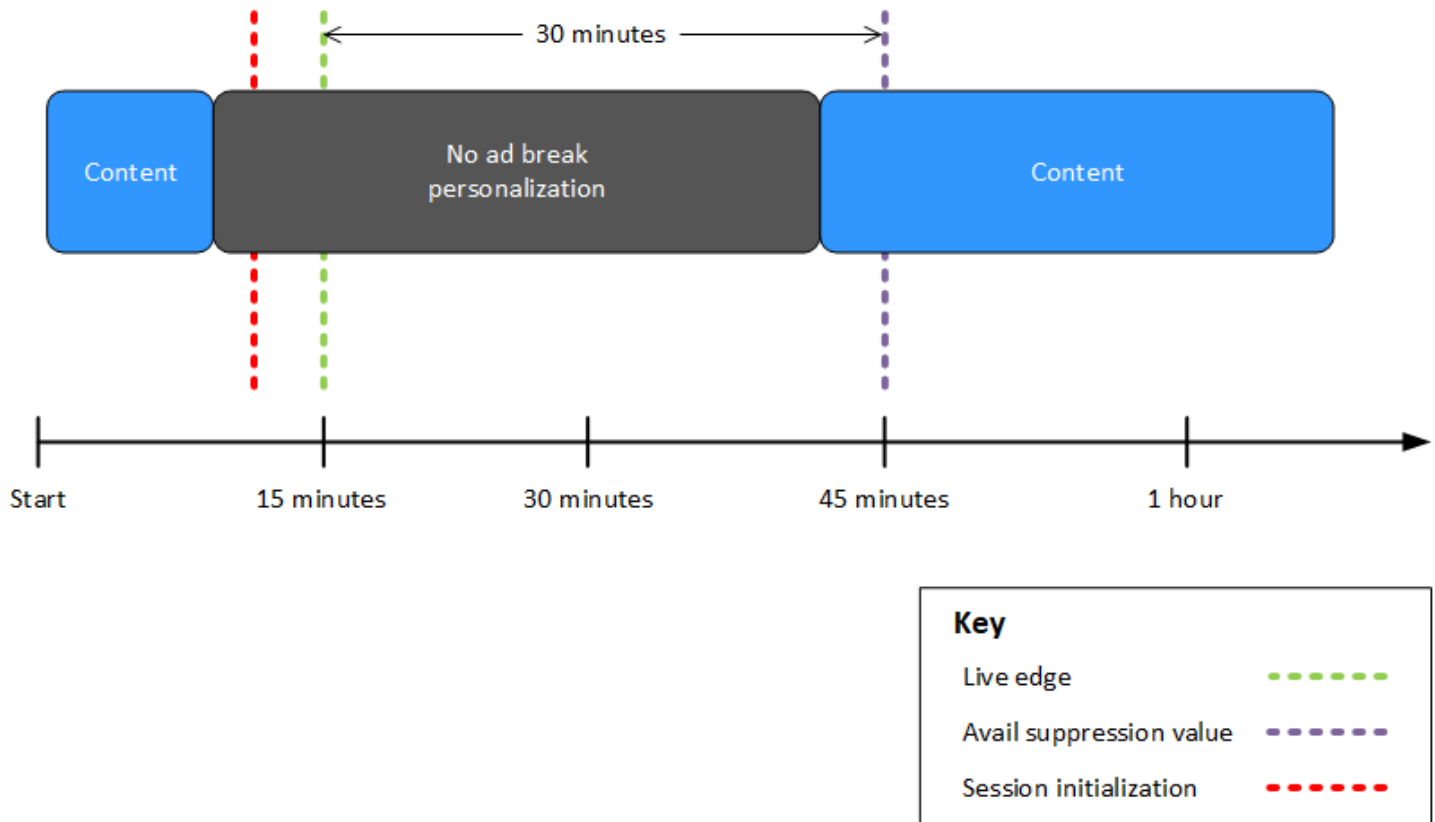
In der folgenden Abbildung sind verschiedene Blöcke horizontal entlang einer Zeitleiste angeordnet, die von links nach rechts verläuft. Jeder Block steht für einen Zeitraum, in dem der Inhalt des Live-Streams oder einer nicht personalisierten Werbepause wiedergegeben wird. Eine gepunktete Linie steht 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 wird von einem Zeitpunkt vor der Sitzungsinitialisierung bis zu einem Zeitpunkt innerhalb des Zeitraums der Verfügbarkeitsunterbrechung ausgeführt.

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. | <ul style="list-style-type: none"> • <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 | <ul style="list-style-type: none"> • <code>PARTIAL_AVAILABLE</code> - nicht verfügbar für den <code>BEFORE_LIVE_EDGE</code> |

| Name | Beschreibung | Akzeptierte Werte |
|------|--|--|
| | aufzurufen, wenn eine Sitzung mitten in der Pause beginnt. | Unterdrückungsmodus <ul style="list-style-type: none"> 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 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 . |

| Abfragezeichenfolgenkomponente | Beschreibung |
|--------------------------------|---|
| = | Ordnet den Parameternamen einem Wert zu. z. B. <code>aws.availSuppressionMode= BEHIND_LIVE_EDGE</code> . |
| & | Verkettet Abfrageparameter. Beispiel: <code>aws.availSuppressionMode= BEHIND_LIVE_EDGE und 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 Wiedergabesitzungsanforderung finden Sie unter [Kundenseitiges Tracking](#).

Beispiel: HLS

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

Stoßstangen einfügen

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 Wiedergabe-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 die 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 Speicherort der gespeicherten Bumper-Asset (s) an.

Beispiel für Start- und End-Bumper-URLs:

Bumper-URL starten: `https://s3.amazonaws.com/startbumperad`

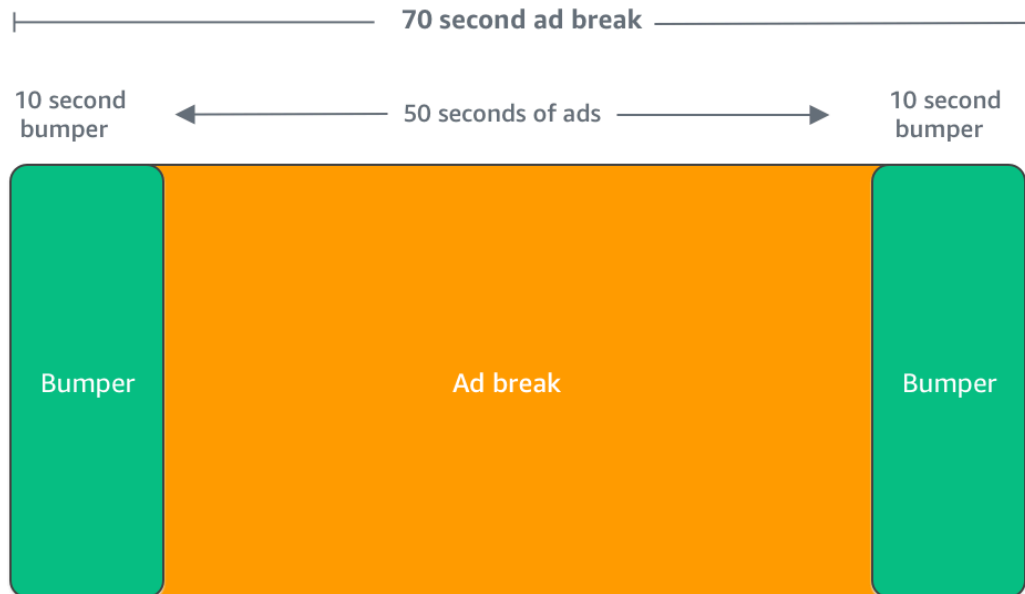
URL für das Ende des Bumpers: `https://s3.amazonaws.com/endbumperad`

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 Werbepause 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

Pre-Roll-Anzeigen sind nur für Live-Workflows verfügbar.

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.

- Die maximal zulässige Dauer von Live-Pre-Roll-Anzeigen ist die maximale Gesamtdauer der Pre-Roll-Anzeigen. MediaTailor greift auf der Grundlage der maximal zulässigen Dauer die folgenden Maßnahmen:
 - Wenn die Gesamtdauer der Anzeigen in der ADS-Antwort unter dem Wert liegt, den Sie unter Höchstdauer des Live-Pre-Rolls 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

Slate ist nur für Live-Workflows verfügbar.

Mit AWS Elemental MediaTailor können Sie eine Slate-Anzeige für Werbeunterbrechungen festlegen. Ein Slate ist ein Standard-MP4-Objekt, 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

Konfiguration des Schiefers

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”](#).

VPAID-Anforderungen

Die Konfiguration eines Pausenfüllers ist erforderlich, wenn Sie VPAID verwenden. MediaTailor Fügt für VPAID das Feld für die Dauer der VPAID-Anzeige ein. In bestimmten Fällen kann diese Dauer etwas länger als die von VAST gemeldete Dauer der VPAID-Werbung sein, um Interaktivität mit Benutzern zu ermöglichen. Der Videoplayer verarbeitet die VPAID-Anzeige dann auf der Grundlage der zurückgegebenen clientseitigen Berichtsmetadaten. MediaTailor Weitere Informationen zur clientseitigen Berichterstattung finden Sie unter [the section called “Kundenseitiges Tracking”](#). Weitere Informationen zu VPAID finden Sie unter [the section called “VPAIDAnforderungen”](#).

Wenn Sie VPAID nicht verwenden und kein Slate konfigurieren, wird MediaTailor standardmäßig der zugrunde liegende Inhaltsstream verwendet.

Anzeigen vorab abrufen

Beim Prefetching werden Anzeigen AWS Elemental MediaTailor proaktiv vom Anzeigenentscheidungsserver (ADS) abgerufen und für bevorstehende Werbeunterbrechungen vorbereitet. Der Vorabruf von Anzeigen trägt dazu bei, die Füllraten von Anzeigen und die Monetarisierung in Live-Workflows mit SCTE -35-Signalen zu maximieren, bei denen es zu Timeouts bei Anzeigenanfragen und bei der Transcodierung kommen kann. Durch das Prefetching von Anzeigen steht mehr Zeit für den programmatischen Anzeigenhandel zur Verfügung. Es reduziert auch die Latenz beim Einfügen von Anzeigen, da sowohl MediaTailor die Transcodierung neuer Inhalte als auch die ADS Antwort im Hintergrund ablaufen.

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

werden müssen. Jeder Prefetch-Zeitplan definiert eine einzelne Gruppe von Anzeigen, die in einer einzelnen MediaTailor Werbepause geschaltet werden sollen. Um Anzeigen für mehrere Werbeunterbrechungen vorab abzurufen, können Sie mehrere Prefetch-Zeitpläne erstellen. Wenn Sie einen Prefetch-Zeitplan erstellen, können Sie Kriterien angeben, mit denen Sie genau steuern können, in welcher Werbepause und in welchem Playback-Stream die vorab abgerufenen Anzeigen platziert werden. MediaTailor

Um Prefetch-Zeitpläne zu erstellen und zu verwalten, können Sie die Konsole oder die verwenden. MediaTailor MediaTailor API

Themen

- [Funktionsweise](#)
- [Prefetch-Zeitpläne erstellen](#)
- [Löschen von Prefetch-Zeitplänen](#)

Funktionsweise

Wenn Ihr Client eine Manifestanforderung an MediaTailor stellt, 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.

Abruf

Dadurch wird das Abrufenfenster definiert. Dabei handelt es sich um den Zeitraum, in dem Anzeigen MediaTailor vorab abgerufen werden. ADS Um das Abrufenfenster einzurichten, legen Sie zunächst fest, wann die Werbeunterbrechung erfolgen soll.

Für fortgeschrittene Anwendungsfälle können Sie optional [dynamische Variablen](#) zur Prefetch-Anfrage hinzufügen, die MediaTailor an die gesendet wird. ADS Auf diese Weise können Sie Sitzungs-, Player- und andere Daten ADS als Teil der Anfrage an die senden. Wenn Sie keine dynamischen Variablen in den Prefetch-Zeitplan aufnehmen, MediaTailor verwendet die dynamischen Variablen, sofern vorhanden, die Sie in Ihren Wiedergabekonfigurationen konfiguriert haben. ADS URL

Verbrauch

Dadurch wird das Nutzungsfenster definiert. Dabei handelt es sich um den Zeitraum, in dem vorab abgerufene Anzeigen in der Werbepause MediaTailor platziert werden.

Für diese Komponente können Sie einem Prefetch-Zeitplan optional bis [zu fünf Auswahlkriterien](#) hinzufügen. MediaTailor verwendet diese Kriterien, um zu bestimmen, ob die Werbepause für die Platzierung der vorab abgerufenen Anzeigen in Frage kommt. Sie können die [scte.event_id](#) dynamische Variable beispielsweise verwenden, wenn Sie möchten, dass der Dienst Anzeigen in einer Werbepause mit einer bestimmten SCTE Event-ID platziert. MediaTailor platziert die vorab abgerufenen Anzeigen nur dann in einem Werbeblock, wenn der Werbeblock die durch die dynamischen Variablen definierten Kriterien erfüllt.

Wenn Ihr Kunde MediaTailor während des Abruffensters Manifestanfragen an sendet, sendet er MediaTailor proaktiv Anfragen an den, die Anzeigen abzurufen und ADS für die spätere Einblendung vorzubereiten. Wenn Sie dynamische Variablen für den Abruf einrichten, MediaTailor werden diese Variablen in die Anfragen aufgenommen.

MediaTailor Findet der Dienst während des Nutzungsfensters eine Markierung von SCTE -35 für Werbeunterbrechungen, verwendet er die Auswahlkriterien, sofern konfiguriert, um zu bestimmen, in welcher Werbeunterbrechung die Werbeanzeigen platziert werden sollen. Wenn die Auswahlkriterien nicht konfiguriert sind, werden die vorab abgerufenen Anzeigen in der ersten Werbepause innerhalb des Nutzungsfensters MediaTailor platziert.

Grundlegendes zu den Kosten für das Prefetching

Für den Abruf von Prefetch-Anzeigen wird Ihnen 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 von finden Sie in der MediaTailor API Referenz. [PrefetchSchedules](#) AWS Elemental MediaTailor API

Note

Wenn Sie in einem Zeitplan Auswahlkriterien für die Verfügbarkeit verwenden möchten, stellen Sie sicher, dass Sie zuerst die ADS URL Vorlage Ihrer Wiedergabekonfiguration mit [dynamischen Variablen](#) konfigurieren. Andernfalls haben die Auswahlkriterien keine Wirkung. Informationen zum Arbeiten mit dynamischen Variablen finden Sie unter [Schritt 3: Konfigurieren Sie die ADS Anfrage URL - und Abfrageparameter](#) dem Thema Erste Schritte beim Einfügen von MediaTailor Anzeigen.

So erstellen Sie mit der Konsole einen neuen Prefetch-Zeitplan

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 dein Origin beispielsweise einen Sport-Stream und einen TV-Show-Stream hat, kannst du die Stream-ID verwenden, um vorab abgerufene Zeitpläne zu erstellen, um Werbeanzeigen einzufügen, die auf den Sport-Stream zugeschnitten sind. Sie übergeben den Stream-ID-Wert an MediaTailor in der Sitzungsinitialisierungs- oder Manifestanfrage Ihres Kunden. 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 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 in den MediaTailor /v1/session-Endpunkt ein. Allgemeine Informationen zur clientseitigen Nachverfolgung finden Sie unter [Kundenseitiges 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. Geben Sie im Bereich Abruf die Abrufeinstellungen an, die Sie verwenden möchten. Diese Einstellungen legen fest, wann Anzeigen aus dem MediaTailor vorab abgerufen werden. ADS Sie bestimmen auch, welche dynamischen Variablen in die Anfrage an die aufgenommen werden sollen ADS, falls vorhanden.
 - Geben Sie unter Startzeit 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 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 im Abschnitt [Dynamische Variablen](#) bis zu 100 dynamische Variablen ein. MediaTailor verwendet diese Variablen als Ersatz in Prefetch-Anfragen, die es an die sendet. ADS Wenn Sie keine dynamischen Variablen eingeben, versucht es nach besten MediaTailor Kräften, die Werte für die in Ihrem enthaltenen dynamischen Variablen zu interpolieren. [ADSURL](#)
 - Wählen Sie Dynamische Variable hinzufügen aus.
 - Geben Sie als Schlüssel einen dynamischen Variablenschlüssel ein, z. `scte.event_id`. B. Sie können jede dynamische Variable verwenden, die MediaTailor unterstützt. Hinweise

zu dynamischen Variablen finden Sie unter [Verwendung dynamischer Anzeigenvariablen in AWS Elemental MediaTailor](#).

- Geben Sie unter Wert einen dynamischen Variablenwert ein, z. B. *my-event*.
 - Um eine weitere dynamische Variable hinzuzufügen, wählen Sie „Dynamische Variable hinzufügen“.
6. 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 dem Prefetch-Verbrauch.
 - 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. Weitere Informationen finden Sie im vorherigen Abschnitt. [avail-matching-criteria](#)
7. 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”](#)

Legen Sie fest, wie oft Ihr Kunde die anrufen soll CreatePrefetchSchedule API

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

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 von finden Sie in der MediaTailor API Referenz. [DeletePrefetchSchedule](#) AWS Elemental MediaTailor API

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 Wiedergabekombi-Konfiguration 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.

Integration einer Inhaltsquelle

In diesem Thema werden die Anforderungen an die Eingabequelle für die MediaTailor Anzeigeneinfügung beschrieben. MediaTailor unterstützt HLS und enthält DASH Eingabequellen.

Themen

- [Anforderungen an die Eingabequelle](#)
- [Eine HLS Quelle integrieren](#)
- [Integration einer MPEG DASH -Quelle](#)

Anforderungen an die Eingabequelle

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

- Verwenden Sie Apple HLS (HTTPLive-Streaming) oder MPEG DASH (Dynamisches Adaptives Streaming überHTTP)
- Verwenden Sie Live-Streaming oder Video-on-Demand (VOD)
- Er muss über das öffentliche Internet zugänglich sein und über eine öffentliche IP-Adresse verfügen.
- Er muss Werbe-Marker in einem der unter [Schritt 2: Bereiten Sie einen Stream vor](#) beschriebenen Formate enthalten.

Eine HLS Quelle integrieren

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 MediaTailor Anzeigeneinblendung verstehen](#).

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

Themen

- [HLSunterstützte Werbemarkierungen](#)
- [Passthrough für Werbemarkierungen aktivieren](#)
- [HLSHandhabung von Manifest-Tags](#)
- [HLSBeispiele für Manifeste](#)

HLSunterstützte Werbemarkierungen

AWS Elemental MediaTailor identifiziert und nutzt Grenzen in einem HLS Manifest und Markierungen im Eingabemanifest. In den folgenden Abschnitten wird beschrieben, welche Marker MediaTailor verwendet werden.

EXT-X- ASSET

Das EXT-X-ASSET Tag enthält Metadaten, die vom Anzeigenentscheidungsserver (ADS) verwendet werden, um Inhalte für den Betrachter zu personalisieren. EXT-X-ASSETParameter sind durch Kommas getrennte Schlüssel-Wert-Paare.

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

- Sie müssen die Werte im URL Ursprungsmanifest -kodieren. *EXT-X-ASSET* Das folgende Beispiel zeigt das EXT-X-ASSET Tag mit Schlüsseln und URL -codierten 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.EPIISODE]&s=[asset.SEASON]&k=[asset.SERIES]
```

Beispiel für VAST eine Anfrage

Das folgende Beispiel zeigt eine VAST GET Anfrage anADS.

```
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- -IN CUE

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-6FFFFFF0",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-6FFFFFF0",START-DATE="2019-01T00:15:00Z\", SCTE35-OUT=0xF
...
#EXT-X-DATERANGE:ID="splice-6FFFFFF0",START-DATE="2019-01T00:15:00Z\", 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-6FFFFFF0",START-DATE="2019-01T00:15:00Z
\",DURATION=60.000,SCTE35-OUT=0xF
...
#EXT-X-DATERANGE:ID="splice-6FFFFFF0",START-DATE="2019-01T00:15:00Z\", SCTE35-
IN=0xF
```

EXT-X- - SPLICEPOINT SCTE35

Sie fügen dem EXT-X-SPLICEPOINT-SCTE35 Werbemarker-Tag eine Nutzlast von SCTE-35 in einer Base64-codierten Binärdatei hinzu. Die dekodierte Binärdatei muss einen Wert von SCTE-35 splice_info_section enthalten, der den Cue-Out-Marker für den Beginn der Anbieterplatzierungs-Opportunity und den 0x34 Cue-In-Marker für das Ende der Anbieterplatzierungs-Opportunity enthält. 0x35

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+tTeaawAiAiBDVUVJAAAKqH+/
DBFNRFNOQjAwMTEzMjIxOTJfTjUAAIiGK1s=
```

Passthrough für Werbemarkierungen aktivieren

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

- EXT-X- -IN CUE
- EXT-X- - CUE OUT
- 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.

HLS Handhabung von Manifest-Tags

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

EXTCUE-X-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- Tags DATERANGE

MediaTailor durchläuft EXT-X-DATERANGE Tags vom Eingabemanifest zum Ausgabemanifest. 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- Tags KEY

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

Unbekannte Tags

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

HLSBeispiele für Manifeste

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

HLSBeispiele für Ursprungsmanifeste

Das folgende Beispiel zeigt ein HLS Master-Manifest, das HLS vom Inhaltsursprung AWS Elemental MediaTailor empfangen wurde.

```
#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 HLS vom Inhaltsursprung AWS Elemental MediaTailor 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
```

HLSBeispiele für personalisierte Manifeste

Das folgende Beispiel zeigt ein HLS Master-Manifest, das AWS Elemental MediaTailor personalisiert wurde.

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-MEDIA:LANGUAGE="eng",AUTOSELECT=YES,FORCED=NO,TYPE=SUBTITLES,URI="../../../../manifest/43f3e412052f2808dd84ea1da90e92e914edddee/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/43f3e412052f2808dd84ea1da90e92e914edddee/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/43f3e412052f2808dd84ea1da90e92e914edddee/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/43f3e412052f2808dd84ea1da90e92e914edddee/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://10380e91fda5e303.mediapackage.us-west-2.amazonaws.com/out/v1/e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779957.ts?m=1566416212
#EXTINF:6.006,
https://10380e91fda5e303.mediapackage.us-west-2.amazonaws.com/out/v1/e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779958.ts?m=1566416212
```

```

#EXTINF:5.372,
https://10380e91fda5e303.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779959.ts?m=1566416212
#EXT-X-DISCONTINUITY
#EXTINF:3.066667,
../..../..../segment/43f3e412052f2808dd84ea1da90e92e914edddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779960
#EXTINF:3.0,
../..../..../segment/43f3e412052f2808dd84ea1da90e92e914edddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779961
#EXTINF:3.0,
../..../..../segment/43f3e412052f2808dd84ea1da90e92e914edddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779962
#EXTINF:3.0,
../..../..../segment/43f3e412052f2808dd84ea1da90e92e914edddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779963
#EXTINF:2.966667,
../..../..../segment/43f3e412052f2808dd84ea1da90e92e914edddee/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779964
#EXT-X-DISCONTINUITY
#EXTINF:6.006,
https://10380e91fda5e303.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779963.ts?m=1566416212
#EXTINF:1.368,
https://10380e91fda5e303.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779964.ts?m=1566416212
#EXTINF:4.638,
https://10380e91fda5e303.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779965.ts?m=1566416212
#EXTINF:6.006,
https://10380e91fda5e303.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779966.ts?m=1566416212
#EXTINF:6.006,
https://10380e91fda5e303.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779967.ts?m=1566416212
#EXTINF:6.006,
https://10380e91fda5e303.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 DASH dynamische Profil entsprechen. MediaTailor akzeptiert Manifesteingaben

für mehrere Perioden und einen Zeitraum und liefert Manifestausgaben, die mit mehreren DASH Perioden DASH kompatibel sind.

Eingabemanifeste müssen Folgendes aufweisen:

- Mindestens ein `Period`-Element mit einem `start`-Attribut.
- SCTE-35 Event-Streams mit Splice-Info-Einstellungen für entweder `splice insert time signal`. Die Einstellungen können im Klartext XML oder in einer Base64-kodierten Binärdatei bereitgestellt werden.
- `Segment templates` mit `segment timelines`.

MediaTailor erfordert für veröffentlichte Manifeste, dass bei Aktualisierungen durch den 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 DASH Manifesten.

Themen

- [DASHWerbemarkierungen](#)
- [DASHund verfügbare Dauer](#)
- [DASHManifest-Segmentnummerierung](#)
- [DASHmanifeste Beispiele](#)
- [DASHStandort-Funktion](#)

DASHWerbemarkierungen

AWS Elemental MediaTailor identifiziert verfügbare Werbeanzeigen in einem DASH Manifest anhand von Markierungen für das Einfügen von Spleißelementen und Time-Signal-Cue-Out wie folgt:

- In einem DASH Manifest mit mehreren Perioden `Period` gilt eine Anzeige als verfügbar, wenn das erste Manifest Event in seinem Event-Stream Spleiß-Insert- oder Time-Signal-Cue-Out-Markierungen enthält. MediaTailor Ignoriert bei mehreren Perioden alle DASH Ereignisse bis auf das erste Ereignis in einer Periode.
- In einem Manifest mit einer einzigen Periode Event gilt ein DASH Manifest als verfügbar, wenn es Spleiß-Insert- oder Time-Signal-Cue-Out-Markierungen enthält.

AWS Elemental MediaTailor Verwaltet Manifeste standardmäßig als Manifeste mit mehreren Perioden DASH. Sie können Ihre Konfiguration so ändern, dass sie DASH Manifeste für einzelne Perioden von Ihrem Ursprungsserver aus verarbeitet. Weitere Informationen finden Sie unter [the section called "Konfiguration erstellen"](#).

Sie können Werbemarkierungen im Klartext XML oder in einer Base64-codierten Binärdatei angeben:

Klar XML

Als Ereignis-Stream-schemeIdUri muss `urn:scte:scte35:2013:xml` eingestellt sein, und für das Ereignis müssen `scte35:SpliceInfoSection`-Marker mit einer der folgenden Einstellungen vorhanden sein:

- `scte35:SpliceInsert` mit der Einstellung `true` für `outOfNetworkIndicator`

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

```
<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>
```

- `scte35:TimeSignal` begleitet von `scte35:SegmentationDescriptor` `scte35:SegmentationUpid`, wobei `segmentationTypeId` auf eine der folgenden Cue-out-Nummern eingestellt ist:
 - 0x22 (Start der Pause)
 - 0x30 (Start der Anbieter-Werbung)
 - 0x32 (Start der Distributor-Werbung)
 - 0x34 (Start der Anbieter-Platzierungsmöglichkeit)
 - 0x36 (Start der Distributor-Platzierungsmöglichkeit)

Das folgende Beispiel zeigt diese Option, wobei die erforderlichen Marker fettgedruckt sind. In diese Beispiel ist als `segmentationTypeId` 52 eingestellt, was 0x34 entspricht.

```
<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>
```

Base64-kodiertes binäres Format

Als Ereignis-Stream-schemeIdUri muss `urn:scte:scte35:2014:xml+bin` eingestellt sein, und für das Ereignis muss `scte35:Signal` `scte35:Binary` mit einem base64-kodierten binären Wert vorhanden sein. Die dekodierte Binärdatei muss die `splice_info_section` gleiche Menge an Informationen enthalten wie die XML Clear-Binärdatei in einem `scte35:SpliceInfoSection` Element. Der Befehlstyp muss entweder `splice_insert()` oder `seintime_signal()`, und die

zusätzlichen Einstellungen müssen den zuvor beschriebenen Einstellungen entsprechen, um eine klare XML Übertragung zu gewährleisten.

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</
Binary>
        </scte35:Signal>
      </Event>
    <Event presentationTime="1541436360" duration="24" id="30">
      <scte35:Signal xmlns="http://www.scte.org/schemas/35/2016">
        <scte35:Binary>QW5vdGhlciB0ZXN0IHN0cmLuZyBmb3IgdW5jb2RpbmcdG8gQmFzZTY0IGVuY29kZWQgYmLuYXJ5Lg=
Binary>
        </scte35:Signal>
      </Event>
```

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": "00:00:24:000000"
      }
    },
    "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
  }
}

```

Bei DASH Manifesten mit mehreren Perioden AWS Elemental MediaTailor wird der erste Wert verwendet, der die Anzeigenplatzierung in einem Event-Stream angibt, und alle weiteren Event Markierungen im Stream werden ignoriert. Berücksichtigt bei DASH Manifesten mit nur einem Zeitraum alle MediaTailor. Event

DASH und verfügbare Dauer

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 Anzeigeneinbindung 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>
    ...
  </EventStream>
</Period>
```

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>
```

DASHManifest-Segmentnummerierung

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>
```

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>

```

DASHmanifeste Beispiele

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

Themen

- [Beispiele für DASH Live-Manifeste](#)
- [VODDASHmanifeste Beispiele](#)

Beispiele für DASH Live-Manifeste

Die folgenden Abschnitte enthalten Beispiele für DASH Live-Manifeste.

Themen

- [DASHBeispiel für das Einfügen von Manifest, Splice](#)
- [DASHBeispiel für ein manifestes Zeitsignal](#)
- [DASHManifest, Base64-kodiertes Binärbeispiel mit einperiodischer Eingabe](#)

DASHBeispiel für das Einfügen von Manifest, Splice

DASHBeispiel für ein Origin-Manifest für das Einfügen von Spleißen

Das folgende Beispiel aus einem MPD Manifest zeigt, dass eine Anzeige in einem Manifest verfügbar ist, das DASH von der Quelle des Inhalts empfangen wurde. Dieses Beispiel verwendet die `scte35:SpliceInsert`-Marker, wobei `outOfNetworkIndicator` auf `true` eingestellt ist.

```

<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>
<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_$.mp4?m=1531257079" initialization="index_audio_2_0_init.mp4?m=1531257079"
startNumber="46042" presentationTimeOffset="7647030507">
            <SegmentTimeline>
                <S t="7647030507" d="168959"/>
                <S t="7647199468" d="176127" r="1"/>
                <S t="7647551723" d="177151"/>
                <S t="7647728875" d="176127" r="1"/>
            </SegmentTimeline>
        </SegmentTemplate>
    </Representation>
    <Representation id="5" bandwidth="128858" audioSamplingRate="44100"
codecs="mp4a.40.2">
        <SegmentTemplate timescale="44100" media="index_audio_4_0_$.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_$.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>
```

```

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

```

DASH-Beispiel für eine personalisierte Antwort für Splice & Insert

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>

```

DASHBeispiel für ein manifestes Zeitsignal

DASHBeispiel für ein Ursprungsmanifest für ein Zeitsignal

Das folgende Beispiel zeigt eine verfügbare Anzeige in einem Manifest, das DASH von der Quelle des Inhalts 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>

```

DASH-Beispiel für eine personalisierte Antwort für ein Zeitsignal

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://d2gh0tfpz97e4o.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"/>
    <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"/>
    </SegmentTimeline>
  </SegmentTemplate>
</Representation>
```

```

        <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>
```

DASHManifest, Base64-kodiertes Binärbeispiel 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 Manifeste mit mehreren Perioden, sowohl für DASH Eingabemanifeste mit mehreren Perioden als auch für Einperioden.

DASHBeispiel für ein Ursprungsmanifest mit einer Periode für eine Base64-kodierte Binärdatei

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>/DAhAAAAAAAAAAP/wEAUAAACIf+9/fgAg9YDAAAAAAAAABiJjIs</
Binary>
      </Signal>
    </Event>
    <Event presentationTime="1550252880" duration="24" id="137">
      <Signal xmlns="http://www.scte.org/schemas/35/2016">
        <Binary>/DAhAAAAAAAAAAP/wEAUAAACJf+9/fgAg9YDAAAAAAAAAC/KdNe</
Binary>
      </Signal>
    </Event>
    <Event presentationTime="1550253000" duration="24" id="138">
      <Signal xmlns="http://www.scte.org/schemas/35/2016">
        <Binary>/DAhAAAAAAAAAAP/wEAUAAACKf+9/fgAg9YDAAAAAAAAADc+01/</
Binary>
      </Signal>
    </Event>
  </EventStream>
  <AdaptationSet...
</AdaptationSet>
</Period>

```

DASH-Beispiel für eine personalisierte Antwort für eine Base64-kodierte Binärdatei mit einer ursprünglichen Manifestkonfiguration mit einer einzigen Periode

Das folgende Beispiel spiegelt die Personalisierung wider, die auf die vorherige Anzeige angewendet wurde AWS Elemental MediaTailor , wenn die MediaTailor Konfiguration Manifeste für einzelne Perioden vom Ursprungsserver angibt. DASH 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"/>
        </SegmentTimeline>
      </Representation>
    </AdaptationSet>
  </Period>
```



```

        <S d="47104" t="74412131853056"/>
        <S d="48128" t="74412131900160"/>
        <S d="48127" t="74412131948288"/>
        <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>
            </Representation>
        </AdaptationSet>
    </AdaptationSet>
</Representation>
</AdaptationSet>

```

```

        </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"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>
</Period>

```

```
<S d="48128" t="74412134588160"/>
<S d="48127" t="74412134636288"/>
<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>
    </AdaptationSet>
</Period>

```



```

        </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/fgAg9YDAAAAAAAAADc+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>
    </EventStream>
  </AdaptationSet>
</Representation>
</AdaptationSet>
</Representation>
</AdaptationSet>

```

```

        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
</Period>

```

VODDASHmanifeste Beispiele

DASHVODManifest der Herkunft

Das folgende Beispiel aus einem MPD Manifest zeigt eine verfügbare Anzeige in einem Video-on-Demand (VOD) -Manifest, das DASH von der Quelle des Inhalts empfangen wurde. Dieses Beispiel verwendet die `scte35:SpliceInsert`-Marker, wobei `outOfNetworkIndicator` auf `true` eingestellt ist.

```

<Period start="PT0.000S" id="8778696" duration="PT29.229S">
  <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2018-07-27T09:35:44.011Z"></SupplementalProperty>
  <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_$.mp4?m=1566416213"
initialization="index_video_7_0_init.mp4?m=1566416213"
startNumber="8778700" presentationTimeOffset="1317997547283">
        <SegmentTimeline>
          <S t="1317997547283" d="180180" r="3"/>
          <S t="1317998268003" d="156156"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation id="2" width="1280" height="720" frameRate="30000/1001"
bandwidth="3299968" codecs="avc1.640029">
      <SegmentTemplate timescale="30000" media="index_video_10_0_$.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>
  </AdaptationSet>
</Period>

```

```

    <Representation id="3" width="640" height="360" frameRate="30000/1001"
    bandwidth="800000" codecs="avc1.4D401E">
      <SegmentTemplate timescale="30000" media="index_video_28_0_${Number}
$.mp4?m=1566416213" initialization="index_video_28_0_init.mp4?m=1566416213"
startNumber="8778700" presentationTimeOffset="1317997547283">
        <SegmentTimeline>
          <S t="1317997547283" d="180180" r="3"/>
          <S t="1317998268003" d="156156"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
  <AdaptationSet mimeType="audio/mp4" segmentAlignment="0" lang="eng">
    <Label>eng</Label>
    <Representation id="4" bandwidth="96636" audioSamplingRate="48000"
codecs="mp4a.40.2">
      <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"></
AudioChannelConfiguration>
      <SegmentTemplate timescale="48000" media="index_audio_5_0_${Number}
$.mp4?m=1566416213" initialization="index_audio_5_0_init.mp4?m=1566416213"
startNumber="8778700" presentationTimeOffset="2108796075909">
        <SegmentTimeline>
          <S t="2108796075909" d="288768"/>
          <S t="2108796364677" d="287744"/>
          <S t="2108796652421" d="288768"/>
          <S t="2108796941189" d="287744"/>
          <S t="2108797228933" d="249856"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation id="5" bandwidth="96636" audioSamplingRate="48000"
codecs="mp4a.40.2">
      <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"></
AudioChannelConfiguration>
      <SegmentTemplate timescale="48000" media="index_audio_8_0_${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"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>

```

```

        <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"></
AudioChannelConfiguration>
    <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>
</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"></SupplementalProperty>
    <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
$.mp4?m=1566416213" initialization="index_video_28_0_init.mp4?m=1566416213"
startNumber="8778705" presentationTimeOffset="1317998424159">
    <SegmentTimeline>
    <S t="1317998424159" d="24024"/>
    <S t="1317998448183" d="180180" r="2"/>
    </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
<AdaptationSet mimeType="audio/mp4" segmentAlignment="0" lang="eng">
    <Label>eng</Label>
    <Representation id="4" bandwidth="96636" audioSamplingRate="48000"
codecs="mp4a.40.2">
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"></
AudioChannelConfiguration>
    <SegmentTemplate timescale="48000" media="index_audio_5_0_$.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"></
AudioChannelConfiguration>
    <SegmentTemplate timescale="48000" media="index_audio_8_0_$Number
$.mp4?m=1566416213" initialization="index_audio_8_0_init.mp4?m=1566416213"
startNumber="8778705" presentationTimeOffset="2108797478789">
        <SegmentTimeline>
            <S t="2108797478789" d="38912"/>
            <S t="2108797517701" d="287744"/>
            <S t="2108797805445" d="288768"/>
            <S t="2108798094213" d="287744"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
<Representation id="6" bandwidth="64643" audioSamplingRate="48000"
codecs="mp4a.40.2">
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"></
AudioChannelConfiguration>
    <SegmentTemplate timescale="48000" media="index_audio_26_0_$Number
$.mp4?m=1566416213" initialization="index_audio_26_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>
</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="8778705" presentationTimeOffset="3953995272478">
    <SegmentTimeline>
    <S t="3953995272478" d="72072"/>
    <S t="3953995344550" d="540540" r="2"/>
    </SegmentTimeline>
    </SegmentTemplate>
  </Representation>
</AdaptationSet>
</Period>

```

DASHVODpersonalisiertes Antwortmanifest

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://10380e91fda5e303.mediapackage.us-west-2.amazonaws.com/out/
v1/5f6a2197815e444a967f0c12f8325a11/</BaseURL>
  <Period duration="PT14.976S" id="8778696_PT0S_0"
  start="PT0S"><BaseURL>https://12345.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_$.mp4" startNumber="1">
    <SegmentTimeline>
    <S d="180000" r="6" t="0"/>
    <S d="87000" t="1260000"/>
    </SegmentTimeline>
    </SegmentTemplate>
</Representation>
    <Representation bandwidth="800000" codecs="avc1.64001e" height="360" id="3"
width="640">
    <SegmentTemplate initialization="asset_360_0_2init.mp4"
media="asset_360_0_2_$.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_$.mp4" startNumber="1" timescale="48000"/>
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="96000"
codecs="mp4a.40.2" id="4">
    <SegmentTemplate initialization="asset_audio_96_3init.mp4"
media="asset_audio_96_3_$.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_$.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://12345.mediatailor.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_$.Number$.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/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="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_${Number$.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_${Number$.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_${Number$.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://12345.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"/>
        <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://12345.mediataylor.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_$.mp4" startNumber="1">
    <SegmentTimeline>
    <S d="180000" r="6" t="0"/>
    <S d="87000" t="1260000"/>
    </SegmentTimeline>
    </SegmentTemplate>
</Representation>
    <Representation bandwidth="800000" codecs="avc1.64001e" height="360" id="3"
width="640">
    <SegmentTemplate initialization="asset_360_0_2init.mp4"
media="asset_360_0_2_$.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_$.mp4" startNumber="1" timescale="48000"/>
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="96000"
codecs="mp4a.40.2" id="4">
    <SegmentTemplate initialization="asset_audio_96_3init.mp4"
media="asset_audio_96_3_$.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_$.mp4" startNumber="1" timescale="48000"/>
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="96000"
codecs="mp4a.40.2" id="5">

```

```

        <SegmentTemplate initialization="asset_audio_96_4_init.mp4"
media="asset_audio_96_4_<Number%09d$.mp4" startNumber="1">
        <SegmentTimeline>
            <S d="96256" r="3" t="0"/>
            <S d="95232" t="385024"/>
            <S d="96256" r="1" t="480256"/>
            <S d="46080" t="672768"/>
        </SegmentTimeline>
    </SegmentTemplate>
</Representation>
</AdaptationSet>
</Period><Period duration="PT5.205S" id="8778696_PT1M23.928S"
start="PT1M23.928S">
    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2018-07-27T09:35:44.011Z"/>
    <AdaptationSet bitstreamSwitching="true" mimeType="video/mp4"
segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
        <Representation bandwidth="2200000" codecs="avc1.640029"
frameRate="30000/1001" height="540" id="1" width="960">
            <SegmentTemplate initialization="index_video_7_0_init.mp4?
m=1566416213" media="index_video_7_0_<Number$.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_<Number$.mp4?m=1566416213"
presentationTimeOffset="1317998268003" startNumber="8778704" timescale="30000">
                <SegmentTimeline><S d="156156" t="1317998268003"/></SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="800000" codecs="avc1.4D401E"
frameRate="30000/1001" height="360" id="3" width="640">
            <SegmentTemplate initialization="index_video_28_0_init.mp4?
m=1566416213" media="index_video_28_0_<Number$.mp4?m=1566416213"
presentationTimeOffset="1317998268003" startNumber="8778704" timescale="30000">
                <SegmentTimeline><S d="156156" t="1317998268003"/></SegmentTimeline>
            </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><Period duration="PT18.818S" id="8778704" start="PT1M29.133S">

```

```

    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2018-07-27T09:36:13.240Z"/>
    <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_$.mp4?m=1566416213"
presentationTimeOffset="1317998424159" startNumber="8778705" timescale="30000">
                <SegmentTimeline>
                    <S d="24024" t="1317998424159"/>
                    <S d="180180" r="2" t="1317998448183"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="3299968" codecs="avc1.640029"
frameRate="30000/1001" height="720" id="2" width="1280">
            <SegmentTemplate initialization="index_video_10_0_init.mp4?
m=1566416213" media="index_video_10_0_$.mp4?m=1566416213"
presentationTimeOffset="1317998424159" startNumber="8778705" timescale="30000">
                <SegmentTimeline>
                    <S d="24024" t="1317998424159"/>
                    <S d="180180" r="2" t="1317998448183"/>
                </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="1317998424159" startNumber="8778705" timescale="30000">
                <SegmentTimeline>
                    <S d="24024" t="1317998424159"/>
                    <S d="180180" r="2" t="1317998448183"/>
                </SegmentTimeline>
            </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_$.mp4?m=1566416213"
      presentationTimeOffset="2108797478789" startNumber="8778705" timescale="48000">
      <SegmentTimeline>
        <S d="38912" t="2108797478789"/>
        <S d="287744" t="2108797517701"/>
        <S d="288768" t="2108797805445"/>
        <S d="287744" t="2108798094213"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>
  <Representation audioSamplingRate="48000" bandwidth="96636"
codecs="mp4a.40.2" id="5">
    <AudioChannelConfiguration
      schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    <SegmentTemplate initialization="index_audio_8_0_init.mp4?
m=1566416213" media="index_audio_8_0_$.mp4?m=1566416213"
      presentationTimeOffset="2108797478789" startNumber="8778705" timescale="48000">
      <SegmentTimeline>
        <S d="38912" t="2108797478789"/>
        <S d="287744" t="2108797517701"/>
        <S d="288768" t="2108797805445"/>
        <S d="287744" t="2108798094213"/>
      </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="2108797478789" startNumber="8778705" timescale="48000">
      <SegmentTimeline>
        <S d="38912" t="2108797478789"/>
        <S d="287744" t="2108797517701"/>
        <S d="288768" t="2108797805445"/>
        <S d="287744" t="2108798094213"/>
      </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="3953995272478" startNumber="8778705" timescale="90000">
      <SegmentTimeline>
        <S d="72072" t="3953995272478"/>
        <S d="540540" r="2" t="3953995344550"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>
</AdaptationSet>
</Period></MPD>
```

DASHStandort-Funktion

Dieser Abschnitt enthält Informationen zur Ortungsfunktion für DASH, die standardmäßig inaktiviert ist AWS Elemental MediaTailor. Lesen Sie diesen Abschnitt, wenn Sie Routingregeln für das Content Delivery Network (CDN) für den Zugriff auf MediaTailor Manifeste erstellen. Lesen Sie diesen Abschnitt auch, wenn Sie serverseitige Berichte für Spieler verwenden, die keine HTTP Sticky-Weiterleitungen unterstützen.

Was ist die Speicherort-Funktion?

Die Standortfunktion ermöglicht es Spielern, die keine HTTP Sticky-Weiterleitungen unterstützen, in ihren Manifest-Aktualisierungsanfragen ein Sticky-Verhalten anzugeben.

AWS Elemental MediaTailor verwendet eine sitzungslöse Initialisierung und verlangt von ihren Spielern ein permanentes HTTP Weiterleitungsverhalten. Bei serverseitiger Berichterstattung gibt der Dienst, wenn der Spieler eine Anfrage für ein Manifest-Update 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. Der Spieler beabsichtigt, dem während der gesamten Sitzung zu folgen, aber Spieler, die HTTP Sticky-Weiterleitungen nicht unterstützen, verwerfen die Weiterleitung und kehren zum Original URL zurück. URL Wenn ein Spieler zur ursprünglichen Sitzung zurückkehrt URL, 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 das Absolute URL in das <Location> Manifest-Tag eingefügt. Spieler, die HTTP Sticky-Weiterleitungen nicht unterstützen, können das URL bereitgestellte in verwenden<Location>, um Aktualisierungen des Manifests anzufordern.

Muss ich die Speicherort-Funktion in meiner Konfiguration deaktivieren?

Die Ortungsfunktion setzt alle CDN Routing-Regeln außer Kraft, die Sie für den Zugriff auf AWS Elemental MediaTailor Manifeste eingerichtet haben, sodass Sie sie möglicherweise deaktivieren müssen. Die Standortfunktion hat keinen Einfluss auf das CDN Zwischenspeichern von Inhalten oder Anzeigensegmenten.

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 Sie keine CDN Routing-Regeln für den Zugriff auf AWS Elemental MediaTailor Manifeste eingerichtet haben, lassen Sie die Standorteinstellung aktiviert.
- Verwenden Sie andernfalls die folgende Regeln:
 - Wenn du entweder keine serverseitige Berichterstattung verwendest oder alle deine Spieler HTTP Sticky-Weiterleitungen unterstützen, deaktiviere die Standortfunktion. Weitere Informationen, wie dies über die Konsole möglich ist, finden Sie unter [the section called "Konfiguration erstellen"](#).
 - Andernfalls wenden Sie sich an den [AWSSupport](#).

Muss die Speicherortfunktion verwendet werden?

Du musst die Standortfunktion für Spieler verwenden, die keine HTTP Sticky-Weiterleitungen unterstützen. Verwende den im <Location> Tag URL angegebenen Wert für all deine Anfragen zur Aktualisierung deiner Manifeste.

Beispiel

Beispiel URLs und <Location> Beispiel-Tag.

- Example Beispiel: Erste Anfrage URL

```
https://b00f3e55c5cb4c1ea6dee499964bea92.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
```

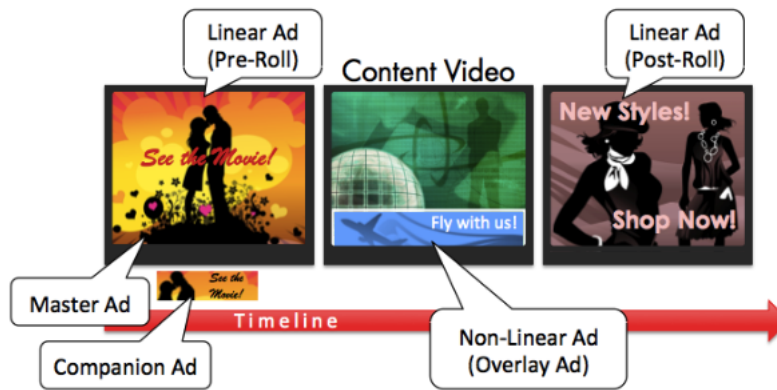
- Example Beispiel: Speicherort-Tag in einem Manifest

```
<Location>https://b00f3e55c5cb4c1ea6dee499964bea92.mediatailor.us-  
east-1.amazonaws.com/v1/dash/5ca4c1892b1f213a1247fad47b3e34c454a7d490/  
testLocationTag/index.mpd?aws.sessionId=0e5d9b45-ae97-49eb-901b-893d043e0aa6</  
Location>
```

Overlay-Anzeigen

Für Live-Streaming-Workflows, bei denen Sie die Monetarisierung erhöhen möchten, ohne das Zuschauererlebnis durch Mid-Roll-Werbung 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“ erscheinen.

MediaTailor erkennt einen SCTE-35-Marker mit Segmentierungstyp als In-Band-Signal für eine Möglichkeit zur Einblendung von Overlay-Werbung. id=0x38 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](#)
- [Erste Schritte](#)
- [Protokollierung und Metriken](#)
- [Fakturierung](#)

Voraussetzungen

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

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](#)
- [Kundenseitiges Tracking](#)

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 `hinzu. off` Der Standardwert ist. `overlays on`
 - (Optional) Geben Sie alle Parameter an, die MediaTailor dann innerhalb eines `adParams` Objekts an das ADS übergeben werden. Diese Parameter entsprechen den

[`player_params.param`] Einstellungen in der ADS-Vorlagen-URL der Konfiguration.
MediaTailor

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 Segmentationsstyp-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,
    "deviceRestrictions": 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:

```
0xfc3035000000000000000000ffff00506fe001b78c8001f021d435545490970d4717fdf00000dbba009076f7665726c617
```

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=0xfc3035000000000000000000ffff00506fe001b78c8001f021d435545490970d4717fdf00000dbba009076f7665726
#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>
<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 `handelt 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/
itrf/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:smpte: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>
```

Kundenseitiges Tracking

MediaTailor platziert die Overlay-Anzeigen, sofern `nonLinearAdsList` sie verfügbar sind. Die MediaTailor clientseitige Tracking-API hat zwei Stammobjekte, die als `avails` und `nonLinearAvails` bezeichnet werden. Wenn es sich bei der VAST-Antwort um eine WMAP mit `breakType of handeltnonlinear`, 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 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

In diesem Abschnitt werden die Protokollierung und die Messwerte für Overlay-Anzeigen erläutert. MediaTailor Weitere Informationen zum Einrichten der Protokollierung finden Sie unter [Überwachung und Tagging](#).

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.

CloudWatchMetriken

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 |
|---|--|
| <code>AdDecisionServer.OverlayAds</code> | Die Anzahl der Overlay-Anzeigen, die innerhalb des von Ihnen angegebenen CloudWatch 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> | <p>Die Anzahl der Avails, die erfolgreich mit mindestens einer Overlay-Anzeige gefüllt wurden:</p> <ul style="list-style-type: none"> • 1 — Es gibt mindestens eine gültige Anzeige. • 0 — Entweder wurden MediaTailor keine Overlay-Anzeigen angezeigt, oder es ist ein anderer Fehler aufgetreten. <p><code>SampleCount</code> verfolgt die Anzahl der ausgefüllten Avails.</p> <p><code>Sum</code> verfolgt die Anzahl der erfolgreich ausgefüllten Overlay-Avails.</p> |
| <code>AdDecisionServer.OverlayMinSuggestedDuration</code> | Die Summe der <code>minSuggestedDuration</code> Dauer aller Anzeigen in Millisekunden, |

| Metrik | Beschreibung |
|---|---|
| | 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. |
| <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. Fakturierung |
| <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> |

Fakturierung

MediaTailor stellt Kunden auf der Grundlage der Anzahl der nichtlinearen Anzeigen in der ADS-Antwort eine Rechnung. Diese Zahl beinhaltet nichtlineare Anzeigen, 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 Startzeit des Videos (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
- Bei clientseitigen Sitzungen, die bereits Wiedergabezeitlinien mit dem erstellen, VST verbessert sich die Sitzung MediaTailor API, da die Sitzung nicht auf den Aufruf des Trackings API angewiesen ist, um die Zeitleiste zu erstellen
- Es ist möglich, sowohl MediaTailor serverseitige Anzeigen 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 das clientseitige Tracking verkaufen. API

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 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 Entscheidungsserver \(ADS\) anzeigen](#)
- [Clientseitiges 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 DASH Playback-PräfixHLS/(implizite Sitzungsinitialisierung) und dem Sitzungsinitialisierungspräfix (explizite Sitzungsinitialisierung).

So aktivieren Sie die Anzeigen-ID für die Sitzung mithilfe der Wiedergabe-Präfixe/HLSDASH

- Initialisieren Sie vom Player aus eine neue MediaTailor Wiedergabesitzung mit einer Anfrage in einem der folgenden Formate, je nach Protokoll:

- Beispiel: Format HLS

```
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
```

Um die Anzeigen-ID für die Sitzung mithilfe des Sitzungsinitialisierungspräfixes zu aktivieren

- 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 übergeben MediaTailor werden sollen. ADS Diese Parameter entsprechen den `[player_params.param]` Einstellungen in der ADS Vorlage 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 Antwort. VAST 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 Erfassung API kreativer Metadaten durch den Kunden und dem Timing/der Positionierung innerhalb des Gesamtzeitplans voneinander zu trennen. Diese Entkopplung reduziert die Wiedergabe-Latenz (insbesondere in VOD Szenarien), in 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 eine VAST Beispielantwort:

```
{
  "version": 1,
  "identifiers": [
    {
      "scheme": "urn:smpte: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-Endpunkt 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 VAST Antwortbeispiel finden Sie im [Interaktionen mit dem Entscheidungsserver \(ADS\) anzeigen](#) Abschnitt.

Personalisierung von HLS Manifesten mit Anzeigenmetadaten

Fügt bei einem HLS Livestream MediaTailor nur Metadaten hinzu, wenn der Stream PROGRAM-DATE-TIME Tags enthält, und zwar mindestens einmal pro Dauer des Manifests. 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 (1970-01-01T00:00:00Z) ist. Wenn das Ursprungsmanifest bereits PROGRAM-DATE-TIME Inhalt enthält, wird dieser MediaTailor Inhalt beibehalten.

MediaTailor personalisiert das Manifest mit den vom Ad Decision Server zurückgegebenen Creatives (). ADS Enthält MediaTailor außerdem für jede Anzeige ein DATERANGE Tag, das sich über die gesamte Laufzeit der Anzeige erstreckt. Das DATERANGE Tag-Format ähnelt dem Format, das im Abschnitt [Werbe-Signalisierung in DASH und HLS in](#) der Version 2023 der SVatechnischen Publikation beschrieben ist.

DasDATERANGE, was MediaTailor generiert wird, hat eindeutige ID-Werte. Um die Eindeutigkeit zu gewährleisten (unter Berücksichtigung der unter [Zuordnung von SCTE -35 zu EXT -X-](#) angegebenen RichtlinienDATERANGE), MediaTailor wird die MEDIA-SEQUENCE Nummer des ersten Anzeigensegments der Avail mit der Sequenznummer der Anzeige innerhalb der Avail verknüpft.

Bei unzureichend ausgefüllten Werbeunterbrechungen in Konfigurationen, bei denen Slate aktiviert ist, werden die Slate-Segmente an das Ende des Avail 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 Ursprung HLS (): **#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:/DA1AAAAAyIYAP/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=/DA1AAAAAyIYAP/wFAUAAAACf
+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_401.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=12.145,Duration=59.993,SCTE35=/DA1AAAAAyIYAP/wFAUAAAACf
+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_402.ts?m=1676054627
```

```
#EXT-X-CUE-OUT-CONT:ElapsedTime=18.151,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/wFAUAAAACf
+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_403.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=24.157,Duration=59.993,SCTE35=/DA1AAAAAyiYAP/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 HLS personalisiertes Manifest (mit kreativer Werbesignalisierung):

Das DATERANGE, was MediaTailor generiert wird, hat eindeutige ID-Werte. Um die Eindeutigkeit zu gewährleisten (unter Berücksichtigung der unter [Zuordnung von SCTE -35 zu EXT -X-](#)

angegebenen Richtlinien(DATERANGE), MediaTailor wird die MEDIA-SEQUENCE Nummer des ersten Anzeigensegments der Avail mit der Sequenznummer der Anzeige innerhalb der Avail verknüpft.

Im folgenden Beispiel wird 421 mit der Nummer der MediaTailor Anzeigenposition verknüpft. 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 VODHLSHerkunft (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 VODHLSHerkunft:

```
#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 VODHLSpersonalisiertes Manifest:

MediaTailor fügt PROGRAM-DATE-TIME VOD Manifeste hinzu, um sie als Anker für die HLS DATERANGE Elemente zu verwenden, die auf Anzeigenpositionen hinweisen.

DasDATERANGE, was MediaTailor generiert wird, hat eindeutige ID-Werte. Um die Eindeutigkeit zu gewährleisten (unter Berücksichtigung der unter [Zuordnung von SCTE -35 zu EXT -X-](#) angegebenen RichtlinienDATERANGE), MediaTailor wird die MEDIA-SEQUENCE Nummer des ersten Anzeigensegments der Avail mit der Sequenznummer der Anzeige innerhalb der Avail verknüpft.

Im folgenden Beispiel wird 421 mit der Nummer der MediaTailor Anzeigenposition verknüpft. 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 DASH von Manifesten mit Anzeigenmetadaten

MediaTailor personalisiert das Manifest mit den vom Ad Decision Server zurückgegebenen Creatives (). ADS 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 SVatechnischen Publikation beschrieben wurde.

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 (SCTEInline-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_$RepresentationID$_
      $Number$.mp4?m=1676062374" initialization="index_video_$RepresentationID$_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_$RepresentationID$_
      $Number$.mp4?m=1676062374" initialization="index_audio_$RepresentationID$_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>
    </AdaptationSet>
  </Period>
</MPD>

```

```

    <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="2023-02-10T21:02:31.007Z"/>
</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_${RepresentationID}_${_}
$Number$.mp4?m=1676062374" initialization="index_video_${RepresentationID}_${_}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_${RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_audio_${RepresentationID$_0_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_$RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_video_$RepresentationID$_0_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 Lineares DASH personalisiertes 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/
f9f38deca3f14fc4b5ab3cdbc76cfb9e/</BaseURL>
  <Location>https://540faac59afd43eeade66624dec85ec7.mediataylor.us-
west-2.amazonaws.com/v1/dash/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/out/v1/
f9f38deca3f14fc4b5ab3cdbc76cfb9e/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_$RepresentationID$_0_init.mp4?
m=1676062374" media="index_video_$RepresentationID$_0_$Number$.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"/>
    </AdaptationSet>
  </Period>
</MPD>

```

```

        <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_${RepresentationID$_0_init.mp4?
m=1676062374" media="index_audio_${RepresentationID$_0_${Number$.mp4?m=1676062374"
presentationTimeOffset="3846790126" startNumber="151" timescale="48000">
            <SegmentTimeline>
                <S d="287744" t="3859981294"/>
                <S d="288768" t="3860269038"/>
                <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://540faac59afd43eeade66624dec85ec7.mediataylor.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:smpte: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://540faac59afd43eeade66624dec85ec7.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:smppte:ul:060E2B34.01040101.01200900.000000000", "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>
    <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://540faac59afd43eeade66624dec85ec7.mediatailor.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:060E2B34.01040101.01200900.000000000","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>
    <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>

```

```

        </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_${RepresentationID$_0_init.mp4?
m=1676062374" media="index_video_${RepresentationID$_0}_${Number$.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_$RepresentationID$_0_init.mp4?
m=1676062374" media="index_audio_$RepresentationID$_0_$Number$.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>
    <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 Entscheidungsserver (ADS) anzeigen

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 ist oder in der VAST Antwort nicht vorhanden ist, MediaTailor wird ein leerer Wert in die Signalisierung der Anzeigen-ID eingefügt.

Example VASTAntwort:

Die folgende VAST Beispielantwort enthält einen `id` Attributwert für den linearen `InLine`-Wert `Creative`. MediaTailor extrahiert den Wert aus dem benutzerdefinierten VAST `Extension` 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>
```

Clientseitiges Tracking API

Das folgende Beispiel zeigt, wie ein Player die Anzeigenmetadaten im Manifest mit den vollständigen Tracking-Eventdaten in der clientseitigen Tracking-Antwort-Payload mit und SDK verknüpft.

`creativeId` `adId`

Example JSONNachricht:

```
{
  "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": []
}
```

Daten zur Berichterstattung und Nachverfolgung

Beacons werden an den Anzeigenserver gesendet, um nachzuverfolgen und zu melden, wie viel von einer Anzeige sich ein Zuschauer angesehen hat. AWS Elemental MediaTailor bietet serverseitige Anzeigenberichte (MediaTailor verfolgt die Anzeige und sendet Beacons) oder clientseitiges Tracking (der Client-Spieler verfolgt die Anzeige und sendet Beacons). Welche Art der Berichterstellung in einer Wiedergabesitzung verwendet wird, ist von der Anforderung abhängig, mit der der Player die Sitzung in MediaTailor einleitet.

Themen

- [Serverseitiges Tracking](#)
- [Kundenseitiges Tracking](#)

Serverseitiges Tracking

Die serverseitige Berichterstellung ist die Standardeinstellung für AWS Elemental MediaTailor. 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, je nach 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 Anforderung mit der Manifest-URL. Das Manifest enthält URLs für die Medien-Manifeste. Die Medien-Manifeste enthalten eingebettete Links für Werbesegment-Anforderungen.

Wenn der Player Wiedergabe von einer Werbesegment-URL (/v1/segment-Pfad) anfordert, sendet AWS Elemental MediaTailor das entsprechende Beacon über die Werbenachverfolgungs-URLs an den Werbeserver. 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 Tracking

Mithilfe des AWS Elemental MediaTailor clientseitigen Trackings API können Sie die Steuerung des Players 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 () und andere Einheiten zur Anzeigenverifizierung. ADS Weitere Informationen zu Impressions- 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 Nachverfolgung können Sie Metadaten an das Wiedergabegerät sendenAPI, sodass zusätzlich zur clientseitigen Nachverfolgung weitere Funktionen verfügbar sind:

Themen

- [Aktivierung der clientseitigen Nachverfolgung](#)
- [Serverparameter hinzufügen](#)
- [Abfrageparameter für Origin-Interaktionen](#)
- [Für die Sitzung konfigurierte Funktionen](#)
- [Bewährte Methoden für die clientseitige Nachverfolgung](#)
- [Schema für kundenseitiges Ad-Tracking](#)
- [Zeitlicher Ablauf von Ad-Tracking-Aktivitäten](#)
- [Spielersteuerung und Funktionalität](#)
- [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 fügt dem Präfix HTTP POST für die Sitzungsinitialisierung der MediaTailor Konfiguration einen Endpunkt hinzu. Optional kann der Spieler 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",
  "overlayAvails":"on"
}
```

'adsParams' is case sensitive
key is not case sensitive
Values can contain spaces. For example, 'value 2' is an allowed value.
this is an example of a query parameter designated for the origin
'overlayAvails' is case sensitive. This is an example of a feature that is enabled at the session level.

Verwenden Sie die MediaTailor Konsole oder konfigurieren Sie API die ADS Anforderungsvorlage soURL, 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 Objekt. `adsParams` JSON In diesem Objekt befinden sich Schlüssel/Wert-Paare, die in allen Sitzungsanfragen gelesen und an den Ad-Server gesendet werden MediaTailor können. MediaTailor unterstützt die folgenden Anzeigenserver:

- Google Ad Manager
- SpringServe
- FreeWheel
- Öffentlich

Abfrageparameter für Origin-Interaktionen

Alle reservierten Schlüssel/Wert-Paare innerhalb der obersten Ebene der JSON Struktur, wie, und `adParams` `availSuppressionOverlayAvails`, werden der ursprünglichen Anfrage URL nicht in Form von Abfrageparametern hinzugefügt. Jede Sitzungsmanifestanforderung, die an den Ursprung MediaTailor gerichtet wird, enthält diese Abfrageparameter. Der Ursprung ignoriert überflüssige Abfrageparameter. MediaTailor Kann beispielsweise die Schlüssel/Wert-Paare verwenden, um Zugriffstoken an den Ursprung zu senden.

Für die Sitzung konfigurierte Funktionen

Verwenden Sie die JSON Sitzungsinitialisierungsstruktur, 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 eine Antwort 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 WerteURLs, 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 für kundenseitiges 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 Workflows beschrieben. VOD

Live-Workflows

Fragen Sie den Tracking-Endpunkt in einem Intervall abHLS, das jeder Zieldauer oder dem minimalen Aktualisierungszeitraum entsprichtDASH, 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.

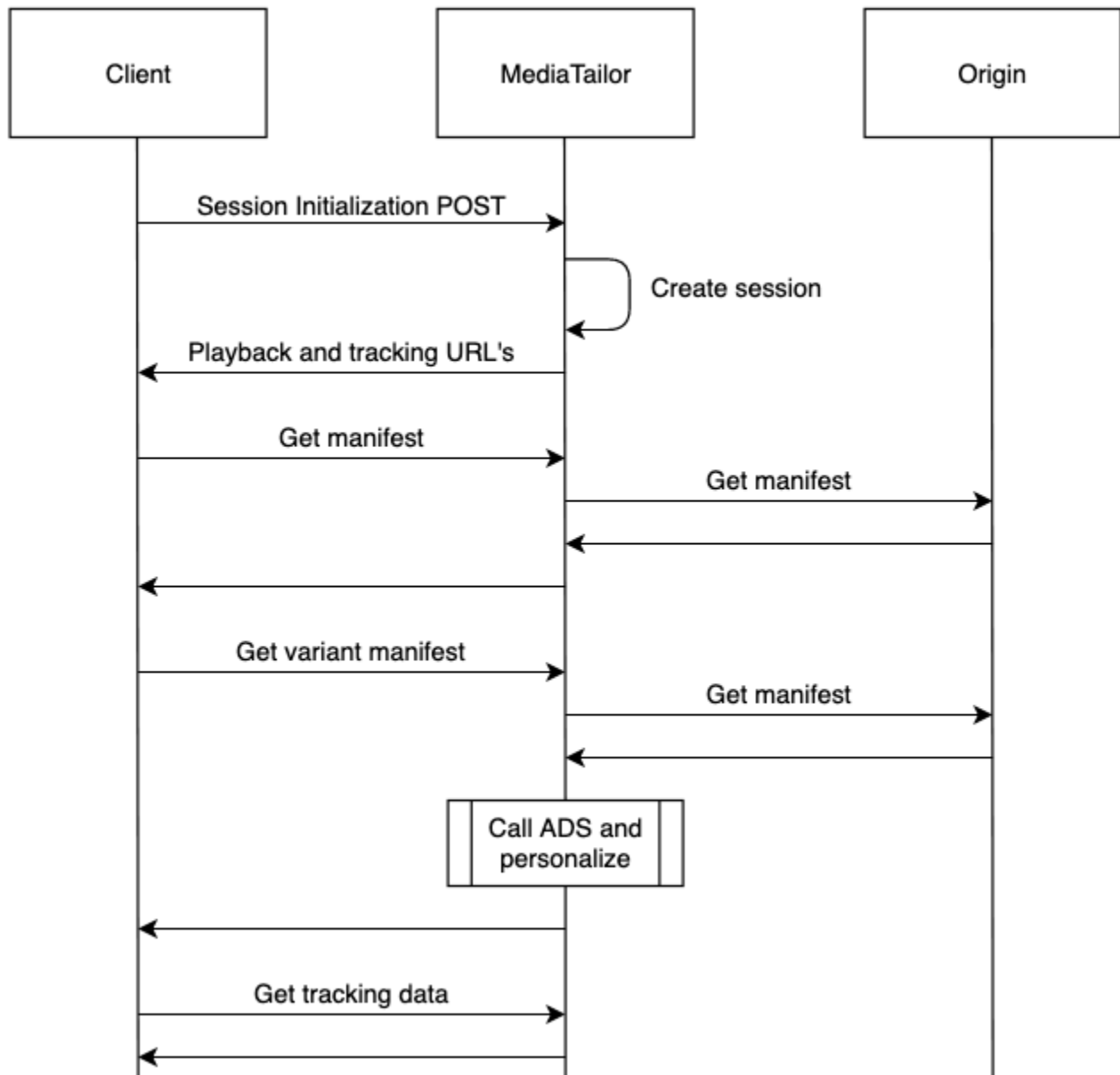
Note

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 jeder verfügbaren Anzeige eine Kennung für den

Zeitraum (DASH) oder das Ereigniselement () zugewiesen. HLS Spieler können diese Manifest-Tags als Aufforderung verwenden, um den MediaTailor Tracking-Endpoint für die Sitzung aufzurufen.

VODArbeitsabläufe

Nach einer erfolgreichen Sitzungsinitialisierung und nach Erhalt des ersten MediaTailor Manifests, das Medien enthält, müssen Sie den Tracking-Endpoint nur einmal aufrufen.



Schema für kundenseitiges Ad-Tracking

In diesem Abschnitt werden Integrationen und das clientseitige Ad-Tracking-Schema beschrieben. MediaTailor erfahren Sie, wie Sie das Schema in eine Player-Umgebung integrieren.

Inhalt


- [Eigenschaften](#)
- [Schema](#)

Eigenschaften

In der folgenden Tabelle sind die Eigenschaften in der clientseitigen Tracking-API, ihre Definitionen, Werttypen und Beispiele aufgeführt.

| Eigenschaft | Definition | Werttyp | Beispiel |
|-----------------------|---|--------------|----------|
| adID | <ul style="list-style-type: none"> • HLS — die Sequenznummer, die dem Anfang der Anzeige zugeordnet ist • DASH — die Perioden-ID der Anzeige | Zeichenfolge | 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. | Zeichenfolge | [] |
| adMarkerDuration | Die anhand der Werbemarkierung im | Zeichenfolge | 30 |

| Eigenschaft | Definition | Werttyp | Beispiel |
|-------------------------------------|--|--------------|-----------------|
| | Manifest beobachtete Gültigkeitsdauer. | | |
| <code>adParameters</code> | Eine Reihe von Anzeigenparametern aus dem VAST VPAID, die an den MediaTailor Player weitergegeben wird. | Zeichenfolge | |
| <code>adProgram DateTime</code> | <ul style="list-style-type: none"> HLS — das Datum im Format ISO/IEC 8601:2004, das die erste Mediensequenz der Anzeige darstellt. GEDANKENS TRICH - | Zeichenfolge | |
| <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 | <code>[]</code> |

| Eigenschaft | Definition | Werttyp | Beispiel |
|-------------|---|--------------|----------|
| adSystem | <p>Der Name des Systems, das die Anzeige bereitstellt.</p> <div data-bbox="472 401 792 1052" style="border: 1px solid #f08080; border-radius: 10px; padding: 10px;"><p> Important Stellen Sie sicher, dass Sie einen Wert angeben. Wenn Sie keinen Wert angeben, können Probleme auftreten.</p></div> | Zeichenfolge | myADS |
| adTitle | Der Titel der Anzeige. | Zeichenfolge | ad1 |

| Eigenschaft | Definition | Werttyp | Beispiel |
|-----------------|---|--------------|-------------------------------------|
| adVerifications | <p>Enthält die Ressourcen 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.</p> <p>MediaTailor unterstützt adVerifications als VAST 3 Erweiterungsknoten.</p> | Array | [] |
| altText | <p>Der alternative Text für ein Bild einer Begleitannonce. Dieser Text ermöglicht es Spielern mit beschreibender Audiounterstützung für sehbehinderte Personen, eine Beschreibung des Bildes vorzulesen.</p> | Zeichenfolge | 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>wierendering Mode</code> „, usw. | Objekt | <code>{}</code> |
| <code>apiFramework</code> | Stellt auf ein, VPAID um dem Player mitzuteilen, dass es sich bei dieser Anzeige um eine VPAID-Anzeige handelt. | Zeichenfolge | VPAID |
| <code>availID</code> | <ul style="list-style-type: none"> • 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. | Zeichenfolge | <ul style="list-style-type: none"> • 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. | Zeichenfolge | |
| <code>bitrate</code> | Die Bitrate des Video-Assets. Diese Eigenschaft ist in der Regel nicht für ein ausführbares Asset enthalten. | Zeichenfolge | <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. | Zeichenfolge | <code>https://aws.amazon.com/</code> |
| <code>companion ClickTracking</code> | Die Tracking-URL für die <code>companion ClickThrough</code> Immobilie. | Zeichenfolge | <code>https://myads.com/beaconing/event=clicktracking</code> |
| <code>creativeId</code> | Der Id Attributwert des Creative Tags für die Anzeige. | Zeichenfolge | <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. | Zeichenfolge | 1 |
| <code>dashAvailabilityStartTime</code> | Bei Live/Dynamic DASH ist dies das MPD@availabilityStartTime Ursprungs-Manifest. | Zeichenfolge | 2022-10-05T19:38:39.263Z |
| <code>delivery</code> | Gibt an, ob ein progressive streaming OR-Protokoll verwendet wird. | Zeichenfolge | 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. | Zahl | 15.015 |

| Eigenschaft | Definition | Werttyp | Beispiel |
|---|--|--------------|-------------------------|
| <code>eventId</code> | <ul style="list-style-type: none"> HLS — die dem Beacon zugeordnete Sequenznummer. DASH — der <code>ptsTime</code> Zeitpunkt des Starts der Anzeige. | Zeichenfolge | 23 |
| <code>eventType</code> | Die Art des Beacons. | Zeichenfolge | <code>impression</code> |
| <code>extensions</code> | Benutzerdefinierte Erweiterungen von VAST, die Werbeserver verwenden. Weitere Informationen zu Erweiterungen finden Sie in Abschnitt 3.18 der VAST 4.2-Spezifikation . | Array | <code>[]</code> |
| <code>height</code> | Die Höhe des Video-Assets in Pixeln. | Zeichenfolge | 360 |
| <code>hlsAnchorMediaSequenceNumber</code> | Die Mediensequenznummer der ersten/ältesten Mediensequenz, die im HLS-Ursprungsmanifest zu finden ist. | Zeichenfolge | 77 |

| Eigenschaft | Definition | Werttyp | Beispiel |
|----------------------------------|--|--------------|--|
| <code>htmlResource</code> | Der CDATA-kodierte HTML-Code, der direkt in die HTML-Seite des Streaming-Anbieters eingefügt wird. | Zeichenfolge | <pre><![CDATA[<!doctype html><html l><head><meta name=\"viewport \" content= \"width=1, initial-s cale=1.0, minimum-s cale=1.0, ...]]></pre> |
| <code>iFrameResource</code> | Die URL zu einer HTML-Ressourcendatei, die der Streaming-Anbieter in einen IFrame lädt. | Zeichenfolge | |
| <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. | Zeichenfolge | <code>https://myad.com/ad/ad134/vpaid.js</code> |

| Eigenschaft | Definition | Werttyp | Beispiel |
|------------------|---|--------------|---|
| mediaType | Der MIME-Typ des kreativen Elements oder des Begleitelements. | Zeichenfolge | video/mp4 |
| meta | | | |
| mezzanine | Die URL des Mezzanine-MP4-Assets, angegeben, falls die VPAID-Anzeige eines enthält. | Zeichenfolge | 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. | Zeichenfolge | UFQzOS44NzNTXzIwMjMtMDctMzFUMTY6NTA6MDYuMzUwNjI2ODQ1Wl8x |
| 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. | Zeichenfolge | 1 |

| Eigenschaft | Definition | Werttyp | Beispiel |
|---------------------------------|--|--------------|--|
| <code>skipOffset</code> | Der Zeitwert, der angibt, wann der Player dem Benutzer Steuerelemente zum Überspringen zur Verfügung stellt. | Zeichenfolge | <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. | Zeichenfolge | <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. | Zahl | <code>9.943</code> |
| <code>staticResource</code> | Die URL zu einer statischen kreativen Datei, die für die Anzeigenkomponente verwendet wird. | Zeichenfolge | <code>https://very-interactive-ads.com/campaign1/file.json?c=1019113602</code> |

| Eigenschaft | Definition | Werttyp | Beispiel |
|-------------|--|--------------|----------|
| vastAdId | Der Id Attributwert des Ad Tags. | Zeichenfolge | ad1 |
| width | Die Breite des Video-Assets in Pixeln. | Zeichenfolge | 640 |

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

| | | | | | |
|--|--------|---|--|--|--|
| - | Objekt | 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 | | | | |

| Schlüssel | Datentyp | Schlüssel für Kinder | Zuordnung von VAST 2.0/3.0 | Kartierung von VAST 4.0 | Beschreibung |
|--|---------------------|--|----------------------------|-------------------------|--|
| <p>MediaTailor erstellt für jede Anzeige innerhalb des Verfügbarkeitszeitraums ein Objekt.</p> | <p>Objekt</p> | <p>adID, AdType, „adParameters,adVerifications,companyAds,duration,durationInSeconds,SymbolExtensions, MediaFile“ dateTimeStart, startTimeInSeconds, adBreakTrackingEvents</p> | | | |
| <p>/adId</p> | <p>Zeichenfolge</p> | | | | <ul style="list-style-type: none"> • 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 | Zeichenfolge | | VAST/Ad/InLine/Creatives/Creative/Linear/AdParameters | | Zeichenfolge mit Anzeigenparametern aus dem VAST-VPAID, 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 erforderlich sind, um die kreative Wiedergabe zu überprüfen |

| 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 Anzeigenerifizierung. | Objekt | Ausführbare Ressourcen-, Lieferanten- und Überprüfungsparameter javascriptResource | | | |
| /executableResource | Array | | VAST/Ad/Inline/AdVerifications/Verification/ExecutableResource | | |
| MediaTailor erstellt ein Objekt für jedes executableResource Element. | Objekt | apiFramework , adType, uri, Sprache | | | |

| Schlüssel | Datentyp | Schlüssel für Kinder | Zuordnung von VAST 2.0/3.0 | Kartierung von VAST 4.0 | Beschreibung |
|---------------|--------------|----------------------|--|--|--------------|
| /apiFramework | Zeichenfolge | | VAST/Ad/Inline/AdVerifications/Verification/ExecutableResource/@apiFramework | | |
| /type | Zeichenfolge | | | | |
| /uri | Zeichenfolge | | VAST/Ad/Inline/AdVerifications/Verification/ExecutableResource/#CDATA | | |
| /language | Zeichenfolge | | 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. | Objekt | apiFramework , Browser (optional), uri | | | |
| /apiFramework | Zeichenfolge | | 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 | Zeichenfolge | | VAST/Ad/Inline/AdVerifications/Verification/JavaScriptResource/#CDATA | | |
| /trackingEvents | Array | | | | |
| MediaTailor erstellt ein Objekt für jeden Tracking-Ereignistyp eines Elements zur Anzeigeverifizierung. | Objekt | Ereignis, URI | | | |

| Schlüssel | Datentyp | Schlüssel für Kinder | Zuordnung von VAST 2.0/3.0 | Kartierung von VAST 4.0 | Beschreibung |
|-----------|--------------|----------------------|--|-------------------------|--------------|
| /event | Zeichenfolge | | 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 | Zeichenfolge | | 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 Begleitanzuzeigen ein Objekt. | Objekt | adParameters , altText , attributes , companionClickThrough , companionClickTracking , htmlResource , Ereignisse verfolgen sequence staticResource | VAST/Ad/InLine/Creatives/Creative/CompanionAds | | |
| /adParameters | Zeichenfolge | | | | |
| /altText | Zeichenfolge | | | | |
| /attributes | Objekt | 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 | Zeichenfolge | | 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 | Zeichenfolge | | 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 Begleitanzuzeigen-Elementen. | | | | | |
| /tracking | Objekt | >Ereignis, URI | | | |
| /event | Zeichenfolge | | 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 | Zeichenfolge | | | | Länge im ISO-8601-Sekunden-Format |
| /durationInSeconds | Zahl | | | | 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 | Objekt | Typ, Inhalt | VAST/Ad/InLine/Extensions/Extension | | |
| /type | Zeichenfolge | | VAST/Ad/InLine/Extensions/Extension/@type | | |
| /content | Zeichenfolge | | | | |
| /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. | Objekt | attributes , dateTime, duration , durationIn Seconds htmlF rce , iconClick s,, iconViewT racking, iFrame ource , staticRes ource startTime startTime InSeconds | VAST/Ad/I nLine/Cre atives/Cr eative/Li near/Icon s/Icon | | |
| /attributes | Objekt | apiFrame ork , durationheight , Offset, Programm, pxratio, XPosition width , YPosition | | | |
| /apiFrame work | Zeichenfolge | | VAST/Ad/I nLine/Cre atives/Cr eative/Li near/Icon s/Icon/@a piFramewo rk | | |

| 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 | Zeichenfolge | | | | |
| /durationInSeconds | Zahl | | | | |
| /htmlResource | Zeichenfolge | | VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/HTMLResource | | |
| /iconClicks | Zeichenfolge | | VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/IconClicks | | |
| /iconClickThrough | Zeichenfolge | | 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 | Objekt | ID | VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/IconClicks/IconClickTracking | | |
| /id | Zeichenfolge | | | | |
| /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 | Zeichenfolge | | 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 | Zeichenfolge | | VAST/Ad/InLine/Creatives/Creative/Linear/Icons/Icon/IconClicks/IconClickFallbackImages/IconClickFallbackImage/@width | | |
| /staticResource | Objekt | 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 | Zeichenfolge | | 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 | Zeichenfolge | | VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/iFrameResource | | |
| /staticResource | Objekt | 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 | Zeichenfolge | | VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/StaticResource/@type | | |
| /uri | Zeichenfolge | | VAST/Ad/Inline/Creatives/Creative/Linear/Icons/Icon/StaticResource/#CDATA | | |
| /startTime | Zeichenfolge | | | | |
| /startTimeInSeconds | Zahl | | | | |

| Schlüssel | Datentyp | Schlüssel für Kinder | Zuordnung von VAST 2.0/3.0 | Kartierung von VAST 4.0 | Beschreibung |
|---|--------------|--|----------------------------|-------------------------|--|
| /mediaFiles | Objekt | adParameters , duration , durationInSeconds , mediaFilesList , mezzanine , Ereignisse verfolgen startTime , startTimeInSeconds | | | Video und andere Inhalte, die der Spieler für die Anzeige benötigt, stehen zur Verfügung |
| /adParameters | Zeichenfolge | | | | |
| /duration | Zeichenfolge | | | | |
| /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 | Zeichenfolge | | | | |
| /delivery | Zeichenfolge | | | | |
| /height | Zeichenfolge | | | | |
| /maintainAspectRatio | Zeichenfolge | | | | |
| /mediaFileUri | Zeichenfolge | | | | |
| /mediaType | Zeichenfolge | | | | |
| /scalable | Zeichenfolge | | | | |
| /width | Zeichenfolge | | | | |
| /mezzanine | Zeichenfolge | | | | |
| /startTime | Zeichenfolge | | | | |
| /startTimeInSeconds | Zeichenfolge | | | | |
| /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 , eventType , startTimeInSeconds | | | |
| /beaconUrls | Array | | | | |
| Eine durch Kommas getrennte Liste aller Tracking-URLs für dieses Ereignis | | | | | |
| /duration | Zeichenfolge | | | | |
| /durationInSeconds | Zahl | | | | |
| /dateTime | Zeichenfolge | | | | |
| /eventId | Zeichenfolge | | | | |
| /eventType | Zeichenfolge | | | | |

| Schlüssel | Datentyp | Schlüssel für Kinder | Zuordnung von VAST 2.0/3.0 | Kartierung von VAST 4.0 | Beschreibung |
|---------------------|--------------|----------------------|----------------------------|-------------------------|---|
| /startTime | Zeichenfolge | | | | |
| /startTimeInSeconds | Zahl | | | | |
| /startTime | Zeichenfolge | | | | Zeitposition im ISO-8601-Sekunden-Format relativ zum Beginn der Wiedergabesitzung |
| /startTimeInSeconds | Zahl | | | | Zeitposition, im Sekundenformat, relativ zum Beginn der Wiedergabesitzung |
| /dateTime | Zeichenfolge | | | | Programmdatum und Uhrzeit im Format ISO 8601 Sekunden für den Start der Anzeige verfügbar |

| Schlüssel | Datentyp | Schlüssel für Kinder | Zuordnung von VAST 2.0/3.0 | Kartierung von VAST 4.0 | Beschreibung |
|----------------------|--------------|----------------------|----------------------------|-------------------------|---|
| /tracking Events | Array | | | | Enthält alle Tracking-Event-Daten, die in der VAST-Antwort empfangen wurden, zusammen mit Zeitinformationen |
| /adType | Zeichenfolge | | | | |
| /availId | Zeichenfolge | | | | |
| /dateTime | Zeichenfolge | | | | |
| /duration | Zeichenfolge | | | | |
| /duration InSeconds | Zahl | | | | |
| /startTime | Zeichenfolge | | | | |
| /startTime InSeconds | Zahl | | | | |

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 z. B. 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.mediatailor.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/ea5485198bf497284559efb8172425e/</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

Spielersteuerung und Funktionalität

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 Werbung \(SIMID\)](#)
- [Interaktive Werbung \(VPAID\)](#)
- [Symbole für Google Warum diese Anzeige \(WTA\)](#)

Schrubben

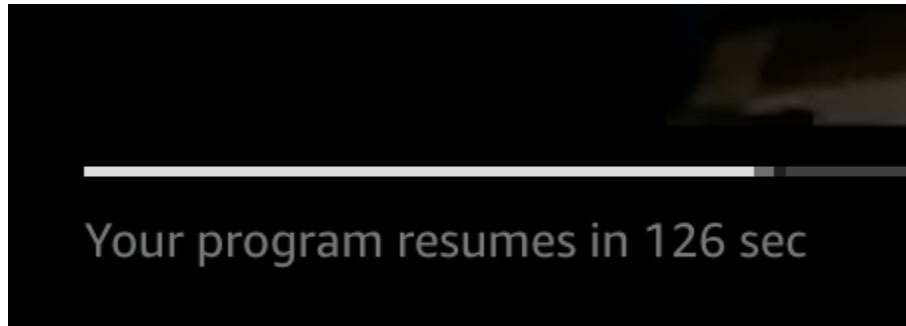
Um das Wiedergabeerlebnis zu verbessern, kann der Player Anzeigenpositionen in der Wiedergabe-Timeline anzeigen. MediaTailor stellt diese Anzeigenpositionen in Form von `adStartTimeInSeconds` Werten in der clientseitigen Tracking-Antwort zur Verfügung.

Note

Einige Streaming-Anbieter verhindern, dass Sie über eine Anzeigenposition hinausgehen.

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 der Countdown erfolgen soll.

Die folgende JSON-Antwort auf die clientseitige Tracking-Payload zeigt die Informationen, die für die Anzeige eines Countdown-Timers für Anzeigen erforderlich sind.

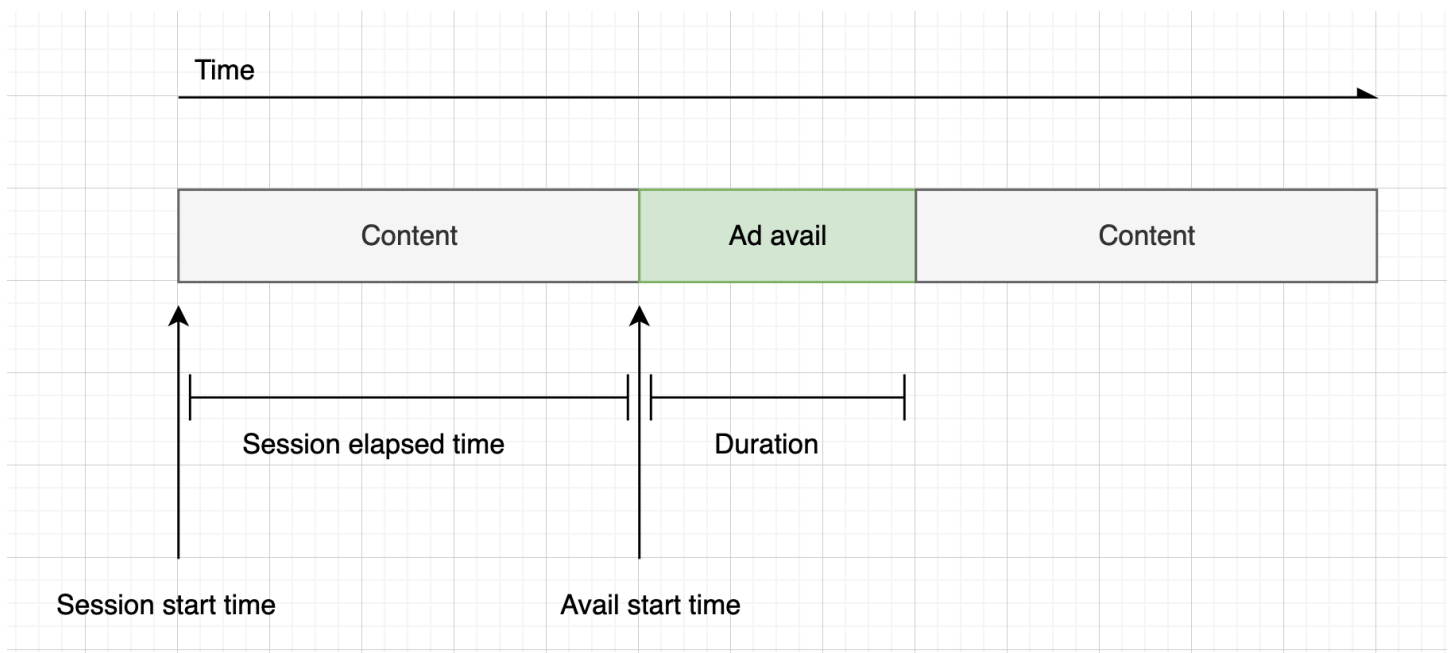
```
{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [...],
      "availId": "7",
      "availProgramDateTime": null,
      "duration": "PT30S",
      "durationInSeconds": 30,
      "meta": null,
      "nonLinearAdsList": [],
      "startTime": "PT28S",
      "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.

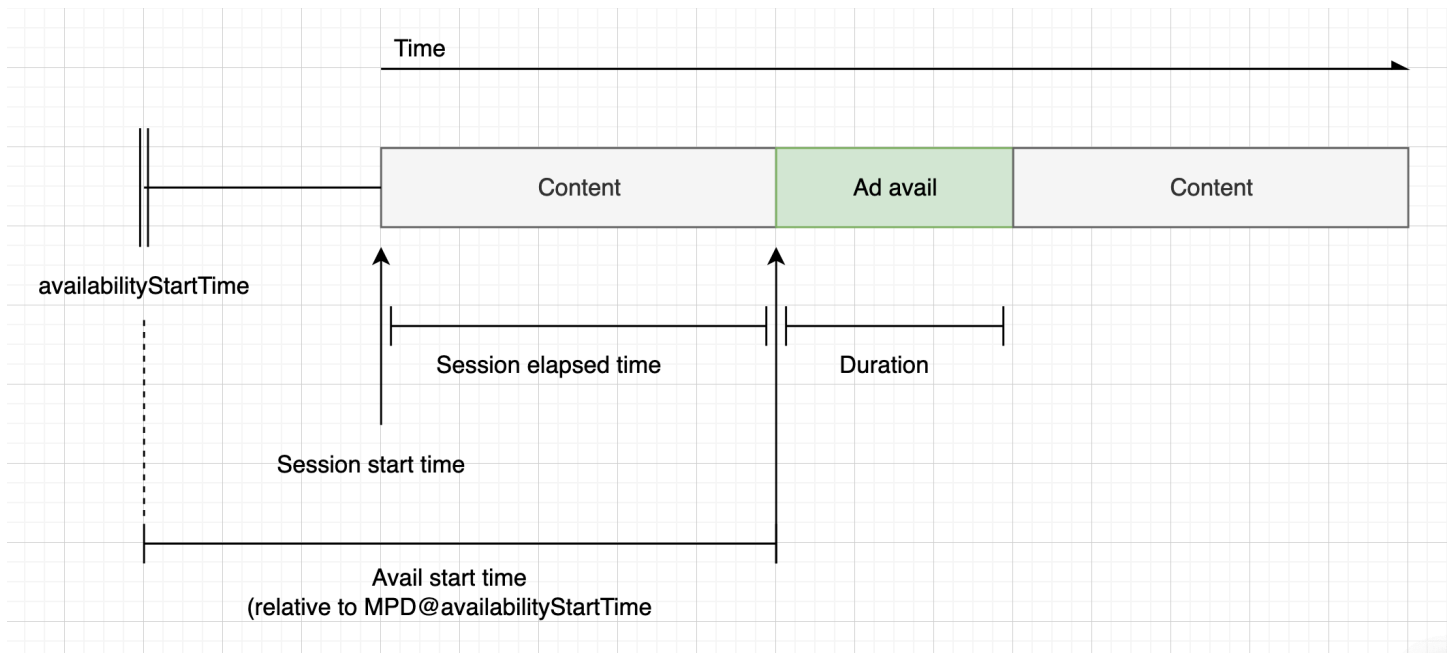
Example Formel: Countdown-Timer für HLS (Live und VOD) und DASH (VOD)

- $session_start_time = \text{die Summe aller EXT-INF Dauerwerte} - \text{der Dauerwert der drei neuesten Mediensequenzen EXT-INF}$
- $Timer\text{-Wert} = duration - (session_elapsed_time - startTime)$



Example Formel: Countdown-Timer für DASH (live)

- $session_start_time = (\text{neuestes Segment } startTime + duration) / timescale - MPD@suggestedPresentationDelay$
- $Timer\text{-Wert} = duration - (session_elapsed_time - startTime)$



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>
```

```
...
</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 alle in der VAST-Antwort zurückgegebenen URLs für lineare Video-Click-Through-Ereignisse analysieren und verfügbar machen. 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 die Click-Through- und Click-Tracking-URLs innerhalb des Arrays MediaTailor angezeigt werden. `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": "UFQz0S4zMz1TXzIwMjMtMDctMzFUMTY6NTQ6MDQu0DA1Mzk2NTI5W18x",
"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

Es hat sich bewährt, dass Anwendungsentwickler eine Logik implementieren sollten, mit der die Begleitanzeige am Ende der Werbeaktion explizit entfernt oder entladen 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",
            }
          ]
        }
      ]
    }
  ]
}
```

```
    "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": "UFQxMFNFmJyAyMy0wNy0wNlQyMToxMDowOC42NzQ4NDA1NjJaXzE%3D",
"nonLinearAvails": []
}
```

Interaktive Werbung (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>
```

```

    </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": "UFQz0S4zMz1TXzIwMjMtMDctMzFUMTY6NTQ6MDQuODA1Mzk2NTI5W18x",
"nonLinearAvails": []
}
```

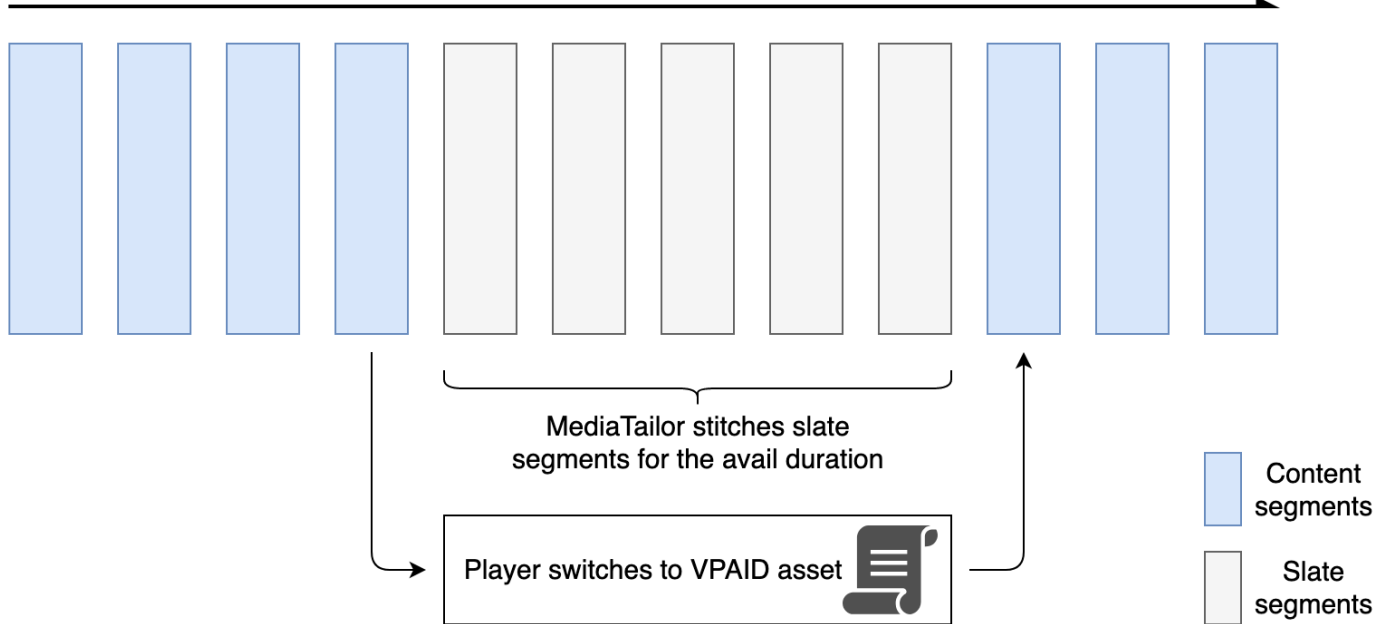
Interaktive Werbung (VPAID)

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.

Timeline



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>
            </TrackingEvents>
          </Linear>
        </Creative>
      </Creatives>
    </Inline>
  </Ad>
</VAST>
```

```

    <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": "UFQz0S4zMz1TXzIwMjMtMDctMzFUMTY6NTQ6MDQuODA1Mzk2NTI5W18x",
"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": "UFQ4TTQyLjI4OVNfMjAyMy0wOC0xNFQx0To1MDo0MS4z0Tc5MjAz0DVaXzE%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 [diesem Beispiel für eine VAST-XML-Antwort](#).

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": "UFQxMFNFmJyMy0wNy0wNlQyMDo0MT0xNy45NDE4MDM0NDhaXzE%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": "PT29.142798675S",
    "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 das Measurement Software Development Kit \(SDK\)](#)
- [Roku Advertising Framework \(RAF\)](#)
- [TheoPlayer](#)
- [MediaTailor Softwareentwicklungskit \(SDK\)](#)

Öffnen Sie das Measurement Software Development Kit (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.

Note

MediaTailor unterstützt derzeit nur VAST Version 3.

Example : Verifizierungsknoten 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": [],
              "javaScriptResource": [
                {
                  "apiFramework": "omid",
                  "browserOptional": "true",

```



```

        "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.

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-Adaptoren für RAF finden Sie unter [Implementieren der serverseitigen Anzeigeneinfügung](#) mithilfe von Roku-Adaptoren 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 der Web- TheoPlayer, Android-, iOS- und tvOS-SDKs finden Sie [MediaTailor](#) auf der Website. MediaTailor TheoPlayer

MediaTailor Softwareentwicklungskit (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 HTML5-basierte Player. 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-Playlisten
- DASH- und HLS-Spezifikationen
- Ereignisbehandlung 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 Werbeanzeigen einzuschränken, die an einen Player zurückgegeben werden. 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 `GetTracking` werden alle möglichen Anzeigen zurückgegeben, die in das Manifestfenster fallen, einschließlich jeweils eines Werts `NextToken` und.
- Wenn eine `GetTracking` Anfrage kein `NextToken` enthält, werden alle Anzeigen im Manifestfenster zurückgegeben.

- Wenn eine `GetTracking` Anfrage ein `NextToken` 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 Anfrage vom Client-Spieler ist eine `GetTracking` Anfrage POST mit einem Anfragetext, der die Werbung `NextToken` und Beacons enthält, die sich auf das Token beziehen.

```
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 Fehlermeldung mit dem HTTP Rückgabecode 400 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, wie in diesem Diagramm dargestellt.

MediaTailor sendet die JSON Antwort `NextToken` wie folgt:

```
{
  "NextToken": "JF57ITe48t1441mv7TmLKuZLroxDzfiSlp6BiSNL1IImzPVMDN0lqrBYycgMbKEb
  "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 bei der nächsten GetTracking Anfrage mit dem NextToken Wert: JF57ITe48t1441mv7TmLKuZLroxDzflslp6BiSNL1IJmzPVMDN 0 lqrBYycg bKEb M.

MediaTailor antwortet mit Anzeigen und Beacons, die StartTimeInSeconds denen entsprechen, die NextToken beim vorherigen Anruf eingestellt wurden.

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 wie in diesem Diagramm dargestellt.

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": ZkfknvbfsgdfbsDFRdffg12EdffecFRvhjyjfhdfhnjtsG5SDGN
  "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
}
]
}
```

Verwendung dynamischer Anzeigenvariablen in AWS Elemental MediaTailor

Die AWS Elemental MediaTailor Anfrage an den Anzeigenentscheidungsserver (ADS) enthält Informationen über die aktuelle Anzeigesitzung, die dem ADS Benutzer helfen, die besten Anzeigen für die Antwort auszuwählen. 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. Zum Beispiel der Antworttyp, der von der MediaTailor erwartet wird ADS.
- Domänenvariablen — Dynamische Variablen, die für URL Domänen verwendet werden können, wie z. B. der my-ads-server.com-Teil von URL <http://my-ads-server.com>. Details hierzu finden Sie unter [Verwenden 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 ihm dabei ADS, zu bestimmen, welche Anzeigen in den Stream eingefügt werden MediaTailor sollen. Details hierzu finden Sie unter [Verwendung von Spielervariablen](#).

Weitere Informationen zur Verwendung dynamischer Domain-, Sitzungs- und Player-Variablen finden Sie unter dem entsprechenden Thema.

Themen

- [Übergeben von Parametern an ADS](#)
- [Fortgeschrittene Nutzung](#)
- [Verwenden von Domänenvariablen](#)
- [Verwenden von Sitzungsvariablen](#)
- [Verwendung von Spielervariablen](#)

Übergeben von Parametern an ADS

Um Sitzungs- und Spielerinformationen an den weiterzuleiten ADS

1. Ermitteln ADS Sie gemeinsam mit dem, welche Informationen er benötigt, damit er auf eine Anzeigenanfrage von antworten kann AWS Elemental MediaTailor.
2. Erstellen Sie eine Konfiguration MediaTailor , die eine ADS Vorlagenanforderung verwendetURL, die die ADS Anforderungen erfüllt. Fügen Sie in URL die statische Parameter und Platzhalter für dynamische Parameter ein. Geben Sie Ihre Vorlage URL in das Feld Ad Decision Server der Konfiguration ein.

In der folgenden Beispielvorlage URL werden Sitzungsdaten und deviceType Spielerdaten bereitgestellt: `correlation`

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

3. Konfigurieren Sie auf dem Player die Sitzungsinitiierungs-Anforderung für AWS Elemental MediaTailor , um Parameter für die Player-Daten bereitzustellen. Sie brauchen die Parameter nur in die Sitzungsinitiierungs-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. In beiden Beispielaufrufen `userID` ist für den ADS und `auth_token` für den Ursprung bestimmt:

- (Option) Aufruf für serverseitiges Ad-Tracking-Reporting — Stellen Sie den Parametern, die Sie an die Datei senden MediaTailor möchten, ein Präfix voran. ADS ads Lassen Sie das Präfix für Parameter, die Sie an den MediaTailor Ursprungsserver senden möchten, aus:

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

HLSBeispiel:

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

DASHBeispiel:

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

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

HLSBeispiel:

```
POST master.m3u8
{
  "adsParams": {
    "deviceType": "ipad"
  }
}
```

DASHBeispiel:

```
POST manifest.mpd
{
  "adsParams": {
    "deviceType": "ipad"
  }
}
```

```
}
```

Wenn der Spieler eine Sitzung initiiert, werden die Variablen in der ADS Vorlagenanforderung URL durch die Sitzungsdaten und die ads Parameter des Players AWS Elemental MediaTailor ersetzt. Die verbleibenden Parameter werden vom Player an den Ursprungs-Server übergeben.

Die folgenden Beispiele zeigen die Aufrufe an den ADS und den Ursprungsserver AWS Elemental MediaTailor, die den Aufrufen des vorherigen Players für die Sitzungsinitialisierung entsprechen:

- 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.

- HLSBeispiel:

```
https://my.origin.server.com/master.m3u8?auth_token=kjhdsaf7gh
```

- DASHBeispiel:

```
https://my.origin.server.com/manifest.mpd?auth_token=kjhdsaf7gh
```

In den folgenden Abschnitten finden Sie Details für das Konfigurieren von Sitzungs- und Player-Daten.

Fortgeschrittene Nutzung

Sie können die ADS Anfrage auf viele Arten mit Spieler- 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 Verketteten 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]
```

Verwenden von Domänenvariablen

Mit dynamischen Domänenvariablen können Sie mehrere Domänen, z. B. den my-ads-server.com-Teil von 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 Anzeigenentscheidungsserver (ADS) in einer einzigen Konfiguration verwenden.

Sie können Domänenvariablen mit jedem Parameter verwenden, der Folgendes enthältURI:

- `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 einen Ursprung 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 Statuscode fehl. HTTP 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` Parameter-Domänen `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 die 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://abc.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 ADSURL, geben Sie in 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 Konfiguration Ihrer ADS Vorlagenanforderung URL 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 For Cable](#).

Weitere Informationen zu Ad Prefetch finden Sie unter [Anzeigen vorab abrufen](#)

| Name | Verfügbar für Ad-Prefetch | SCTEAbschnitt mit den Spezifikationen von -35 | 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 sie gestellt wird, enthält diese die Indexnummer der Anzeigenverfügbarkeit. Dieser Parameter ermöglicht es Ihnen, die Anzeigenauswahl zu verbessern, indem Sie Funktionen wie Wettbewerbsausschluss und Frequenzbegrenzung nutzen. |
| [avail.random] | Ja | | Eine Zufallszahl zwischen 0 und 10.000.000.000 als lange Zahl, die für jede Anfrage an den 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.avail_num] | Ja | 9.7.2.1 | Der Wert, nach dem MediaTailor aus dem Feld SCTE -35 geparkt 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 | SCTEAbschnitt mit den Spezifikationen von -35 | Beschreibung |
|--|---------------------------|---|--|
| [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.del ivery_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.dev ice_restr ictions] | 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 -35-Spezifikation. SCTE |
| [scte.eve nt_id] | Ja | 9.1 und 9.7.2.1 | Der Wert, nach dem MediaTailor aus dem Feld SCTE -35 geparkt wurde <code>splICE_event_id</code> , als lange Zahl. MediaTailor verwendet diesen Wert, um lineare Anzeigenverfügbarkeitsnummern zu kennzeichnen oder um Abfragezeichenfolgen für Anzeigenserver, wie z. B. die Positionen von Anzeigenpods, aufzufüllen. |
| [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 | SCTEAbschnitt mit den Spezifikationen von -35 | 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 -35-Spezifikation. SCTE |

| Name | Verfügbar für Ad-Prefetch | SCTEAbschnitt mit den Spezifikationen von -35 | Beschreibung |
|---------------------------------|--|---|---|
| [scte.segmentation_upid] | segmentation_upid_type : Yes private_data : Yes | segmentation_upid: 10.3.3.1 Privat verwaltet : UPID 10.3.3.3 | <p>Entspricht dem Element SCTE segmentation_upid -35. Das segmentation_upid Element enthält segmentation_upid_type und segmentation_upid_length .</p> <p>MediaTailor unterstützt die folgenden segmentation_upid Typen:</p> <ul style="list-style-type: none"> • ADSInformation (0x0E) — Werbeinformationen. Weitere Informationen finden Sie in der Beschreibung segmentation_upid in der -35-Spezifikation. SCTE • Managed Private UPID (0x0C) — Die Managed Private () -Struktur, wie sie in der UPID -35-Spezifikation MPU definiert ist. SCTE MediaTailor unterstützt Binärdateien oder Repräsentationen. DASH XML SCTE <p>Sie können diese Struktur in einem Podbuster-Workflow verwenden. Geben Sie dazu einen 32-Bit-Wert (4 Byte) an und format_identifizieren Sie die folgenden Parameter in das private_data Attribut auf:</p> <pre data-bbox="764 1503 1507 1663">ABCD{"assetId": " my_program ", "cueData": {"cueType": " theAdType ", "key": " pb", "value": " 123456"}}</pre> <p>MediaTailor analysiert die Werte aus dem vorherigen JSON und übergibt sie an die scte.segmentation_upid.cueData.value</p> |

| Name | Verfügbar für Ad-Prefetch | SCTEAbschnitt mit den Spezifikationen von -35 | Beschreibung |
|---|---------------------------|---|---|
| | | | <p>dynamischen Variablen <code>scte.segmentation_upid.assetId</code> <code>scte.segmentation_upid.cueData.key</code> , und.</p> <ul style="list-style-type: none"> • Benutzerdefiniert (0x01) — Eine benutzerdefinierte Struktur. Weitere Informationen finden Sie in der Beschreibung <code>segmentation_upid</code> in der -35-Spezifikation. SCTE |
| [scte.segmentation_upid.assetId] | Ja | | <p>Wird in Verbindung mit Managed Private UPID (0xC) für Podbuster-Workflows verwendet. <code>segmentation_upid_type</code> MediaTailorleitet diesen Wert aus dem <code>assetId</code> Parameter in der <code>s</code>-Struktur ab. MPU <code>private_data</code> JSON Weitere Informationen finden Sie unter Managed Private UPID JSON structure for a podbuster workflow.</p> |
| [scte.segmentation_upid.cueData.key] | Ja | | <p>Wird in Verbindung mit Managed Private UPID (0xC) <code>segmentation_upid_type</code> für Podbuster-Workflows verwendet. MediaTailorleitet diesen Wert aus dem <code>cueData.key</code> Parameter in der <code>s</code>-Struktur ab. MPU <code>private_data</code> JSON Weitere Informationen finden Sie unter Managed Private UPID JSON structure for a podbuster workflow.</p> |
| [scte.segmentation_upid.cueData.value] | Ja | | <p>Wird in Verbindung mit Managed Private UPID (0xC) <code>segmentation_upid_type</code> für Podbuster-Workflows verwendet. MediaTailorleitet diesen Wert aus dem <code>cueData.key</code> Parameter in der <code>s</code>-Struktur ab. MPU <code>private_data</code> JSON Weitere Informationen finden Sie unter Managed Private UPID JSON structure for a podbuster workflow.</p> |

| Name | Verfügbar für Ad-Prefetch | SCTEAbschnitt mit den Spezifikationen von -35 | Beschreibung |
|-------------------------------------|---------------------------|---|---|
| [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 segments_expected 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 sub_segment_num in der -35-Spezifikation. SCTE |
| [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 sub_segments_expected in der -35-Spezifikation. SCTE |
| [scte.unique_program_id] | Ja | 9.7.2.1 | Der Integer-Wert, nach dem MediaTailor aus dem Feld -35 geparkt wurde. SCTE splice_insert unique_program_id Der ADS verwendet die eindeutige Programm-ID (UPID), um gezielte Werbung auf Programmebene für lineare Livestreams bereitzustellen. Wenn der Befehl SCTE -35 nicht Splice Insert ist, wird dieser Wert auf einen MediaTailor leeren Wert gesetzt. |

| Name | Verfügbar für Ad-Prefetch | SCTEAbschnitt mit den Spezifikationen von -35 | Beschreibung |
|--------------------------------------|---------------------------|---|--|
| [session.avail_duration_ms] | Ja | | <p>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:</p> <ul style="list-style-type: none"> • ForHLS: MediaTailor Ruft die Dauer aus den Werten <code>#EXT-X-CUE-OUT: DURATION</code> oder aus den Werten im <code>#EXT-X-DATERANGE</code> Tag ab. Wenn das Eingabemanifest eine Dauer von Null, ungültig oder 0 für die in diesen Tags verfügbare Anzeige hat, wird der Standardwert MediaTailor verwendet. • ForDASH: MediaTailor Ruft den Wert für die Dauer aus der Eventdauer ab, sofern eine angegeben wurde. Andernfalls wird der Standardwert verwendet. • FürVOD: Wenn ein VOD Stream einen Pre-Roll-Anzeigenaufruf auslöst und das Manifest keine SCTE Nachrichten mit einem Wert für die Dauer enthält, wird MediaTailor keine Dauer für <code>[session.avail_duration_ms]</code> eingegeben, einschließlich des Standardwerts für die Dauer. |
| [session.avail_duration_secs] | Ja | | <p>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 <code>[session.avail_duration_ms]</code>.</p> |

| Name | Verfügbar für Ad-Prefetch | SCTEAbschnitt mit den Spezifikationen von -35 | Beschreibung |
|-----------------------------|---------------------------|---|---|
| [session.client_ip] | Nein | | Die Remote-IP-Adresse, von der die MediaTailor Anfrage kam. Wenn der <code>X-forwarded-for</code> Header gesetzt ist, wird dieser Wert für MediaTailor verwendet <code>client_ip</code> . |
| [session.id] | Nein | | Eine eindeutige numerische Kennung für die aktuelle Wiedergabebesitzung. Alle Anfragen, die ein Spieler für eine Sitzung stellt, haben dieselbe ID, sodass sie für ADS Felder verwendet werden kann, die Anfragen für eine einzelne Wiedergabe korrelieren sollen. |
| [session.referrer] | Nein | | Normalerweise die Seite, auf URL der der Videoplayer gehostet wird. MediaTailor setzt diese Variable auf den Wert des <code>Referer</code> Headers, den der Player in seiner Anfrage verwendet hat MediaTailor. Wenn der Spieler diesen Header nicht bereitstellt, MediaTailor lässt [session.referrer] er 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 <code>User-Agent</code> Header, der von der Sitzungsinitialisierungsanfrage des Players MediaTailor empfangen wurde. Wenn Sie einen CDN OR-Proxy vor dem Manifest-Endpunkt verwenden, müssen Sie hier den richtigen Header des Players als Proxy verwenden . |

| Name | Verfügbar für Ad-Prefetch | SCTEAbschnitt mit den Spezifikationen von -35 | Beschreibung |
|-----------------------|---------------------------|---|--|
| [session.uuid] | Nein | | Alternative zu [session.id] . Hierbei handelt es sich um eine eindeutige Kennung für die aktuelle Wiedergabebesitzung, wie z. B. im Folgenden: <div style="border: 1px solid #ccc; border-radius: 10px; padding: 5px; margin-top: 10px; text-align: center;"> e039fd39-09f0-46b2-aca9-9871cc116cde </div> |

Example

Wenn für ein benannter `deviceSession` Abfrageparameter mit der eindeutigen Sitzungs-ID übergeben werden muss, AWS Elemental MediaTailor könnte die darin ADS URL enthaltene Vorlage wie folgt aussehen: ADS

```
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 Kennung lautet `1234567`, ADS würde die letzte Anfrage, die MediaTailor an die gestellt wird, in etwa so aussehen:

```
https://my.ads.server.com/path?deviceSession=1234567
```

Wenn für mehrere Abfrageparameter übergeben werden müssen, AWS Elemental MediaTailor könnte die Vorlage ADS URL wie folgt aussehen: ADS

```
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 DASH XML Marker-Beispielfragment zeigt, wie man es benutzt `scte35:SpliceInsert`:

```
<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>
  </Event>
</EventStream>
```

Das folgende DASH XML Marker-Beispielfragment zeigt, wie man es benutzt `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>
  </EventStream>
</Period>
```

Das folgende DASH XML Marker-Beispielfragment zeigt, wie man es benutzt `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>QW5vdGhlciB0ZXN0IHN0cmVuZyBmb3IgaW5jb2RpbmcdG8gQmFzZTY0IGVuY29kZWQgYmLuYXJ5Lg=
Binary>
  </scte35:Signal>
</Event>
```

Das folgende HLS Tag-Beispiel zeigt die Verwendung von EXT-X-DATERANGE:

```
#EXT-X-DATERANGE:ID="splice-6FFFFFF0",START-DATE="2014-03-05T11:
15:00Z",PLANNED-DURATION=59.993,SCTE35-OUT=0xFC002F0000000000FF0
00014056FFFFFFFF00E011622DCAFF000052636200000000000A0008029896F50
0000087000000000
```

Das folgende HLS Tag-Beispiel zeigt, wie Folgendes verwendet wird EXT-X-CUE-OUT:

```
#EXT-0ATCLS-SCTE35:/DA0AAAAAAAAAAAAABQb+ADAQ6QAeAhxDVUVJQAAA03/PAAEUrEoICAAAAAg
+2UBNAAANvrtoQ==
#EXT-X-ASSET:CAID=0x0000000020FB6501
#EXT-X-CUE-OUT:201.467
```

Das folgende HLS Tag-Beispiel zeigt, wie Folgendes verwendet wird EXT-X-SPLICEPOINT-SCTE35:

```
#EXT-X-SPLICEPOINT-SCTE35:/DA9AAAAAAAAAAP/wBQb+uYbZqwAnAiVDVUVJAAAKqX//
AAEjW4AMEU1EU05CMDAxMTMyMjE5M190NAAAmXz5JA==
```

Das folgende Beispiel zeigt, wie scte35:Binary decode 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": "00:00:24:00000"
    }
  },
  "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 die gesendet werden ADS ADSURL, geben Sie in 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 [`player_params.user_id`] in die ADS URL Konfiguration einzubeziehen.

Auf diese Weise können Sie die Abfrageparameter steuern, die in der ADS Anfrage enthalten sind. In der Regel fügen Sie der ADS Anforderung einen speziellen Abfrageparameter hinzu, den er ADS erkennt, URL 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 AWS Elemental MediaTailor die ADS Anforderungsvorlage so, URL dass sie auf die Parameter verweist. Im Folgenden URL wird die Aufnahme der Beispielpaare veranschaulicht:

```
https://my.ads.com/path?param1=[player_params.param1]&param2=[player_params.param2]
```

2. (Optional) Für serverseitige Ad-Tracking-Berichte kodieren Sie die URL Schlüssel-Wert-Paare auf dem Player mit `-`codieren. Wenn MediaTailor er die Anfrage zur Sitzungsinitialisierung erhält, URL dekodiert er die Werte einmal, bevor sie in die Anfrage eingefügt werden. ADS URL

 Note

Wenn Sie einen URL `-`codierten Wert ADS benötigen, URL kodieren Sie den Wert zweimal auf dem Player. Auf diese Weise MediaTailor ergibt die Dekodierung von einen einmal codierten Wert für. ADS

Wenn zum Beispiel die dekodierte Darstellung der an den gesendeten Werte ADS `istparam1=value1:¶m2=value2:`, dann ist es die `-`kodierte Darstellung. URL `param1=value1%3A¶m2=value2%3A`

3. Übergeben Sie im Sitzungsinitialisierungsaufwurf des Players 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.

- Beispielanfragen für serverseitige Ad-Tracking-Berichte — unter Verwendung von -codierten Paaren URL

HLS:

```
<master>.m3u8?ads.param1=value1%3A&ads.param2=value2%3A
```

DASH:

```
<manifest>.mpd?ads.param1=value1%3A&ads.param2=value2%3A
```

- Beispielanfrage für clientseitige Ad-Tracking-Berichte — ohne -Kodierung URL

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 Player-Anfrage empfangen wird. Bei der clientseitigen Berichterstattung werden die in der Payload empfangenen Parameter nicht geändert. JSON MediaTailor sendet die folgende Anfrage an: ADS

```
https://my.ads.com/<path>?param1=value1:&param2=value2:
```

Auf diese Weise werden die `param1 param2` Schlüssel-Wert-Paare als erstklassige Abfrageparameter in die ADS Anfrage aufgenommen.

Arbeiten mit CDNs

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, stellen sich Anforderungs- und Antwort-Ablauf wie folgt dar:

1. Der Player 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 Player fordert Segmente von den URLs an, die im Manifest bereitgestellt werden.
4. Das CDN übersetzt die Segment-URLs. Es leitet Anfragen zu Inhaltssegmenten an den Ursprungsserver und Anzeigenanfragen an die CloudFront Amazon-Distribution weiter, in MediaTailor der transkodierte Anzeigen gespeichert werden.
5. Der Ursprungsserver MediaTailor antwortet mit den angeforderten Segmenten, und die Wiedergabe beginnt.

In den folgenden Abschnitten wird beschrieben, wie AWS Elemental MediaTailor und das CDN zum Ausführen dieses Ablaufs konfiguriert werden.

Integration eines CDN

In den folgenden Schritten wird gezeigt, wie AWS Elemental MediaTailor in Ihr Netzwerk zur Bereitstellung von Inhalten (Content Distribution Network, CDN) integriert wird. 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 Anfragen nach Anzeigensegmenten an die interne CloudFront Amazon-Distribution weiterleitet, in der 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 AWS Elemental MediaTailor eine Standardkonfiguration bereitstellt.

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 der CDN-Weiterleitungsverhaltensweisen dem Ursprungs-Server und dem Werbe-Speicherort zuweist. Geben Sie die Domänennamen 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-Domäne.

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 Namen des Verhaltens.

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

Durch Verwendung eines CDN für Manifest- und Berichtsabfragen 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 Domänennamen 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 der serverseitigen Berichterstellung lässt sich durch Verweisen auf ein CDN vor `/v1/segment` in Werbesegment-Anforderungen verhindern, dass AWS Elemental MediaTailor Werbenachverfolgungs-Beacons doppelt sendet. 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.

Wenn Sie diese Vorteile nutzen möchten, erstellen Sie im CDN Verhaltensweisen, mit denen Anforderungen an den AWS Elemental MediaTailor-Konfigurationsendpunkt weitergeleitet werden. 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://a57b77e98569478b83c10881a22b7a24.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://a57b77e98569478b83c10881a22b7a24.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://a57b77e98569478b83c10881a22b7a24.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://a57b77e98569478b83c10881a22b7a24.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

- Player-Anforderungen an `https://CDN_Hostname/some/path/asset.m3u8` werden, basierend auf dem Schlüsselwort `*.m3u8` in der Anforderung, an den AWS Elemental MediaTailor-Pfad `https://mediatailor.us-west-2.amazonaws.com/v1/session/configuration/endpoint` weitergeleitet.
- Player-Anforderungen an `https://CDN_Hostname/some/path/asset.mpd` werden, basierend auf dem Schlüsselwort `*.mpd` in der Anforderung, an den AWS Elemental MediaTailor-Pfad `https://mediatailor.us-west-2.amazonaws.com/v1/dash/configuration/endpoint` weitergeleitet.

Wie AWS Elemental MediaTailor geht man mit BaseURLs für DASH 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 der im Manifest angegebenen URLs. 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, 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.

Für Werbesegmente verfährt AWS Elemental MediaTailor wie folgt:

- 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, fügt MediaTailor 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.

Das Verhalten bei der 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.

- MediaTailor Ersetzt beim Ersetzen von Anzeigen Inhaltssegmente durch Anzeigen.
- 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, wenn Sie das entsprechende Thema auswählen.

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-Werbemarkierungen 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 einen Post-Roll-Avail wie den folgenden 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. Werbemarkierungen sind jedoch nicht erforderlich. Wenn das Manifest keine Werbe-

Marker enthält, macht MediaTailor einen einzigen Aufruf an den ADS und erstellt Werbepausen basierend auf der Antwort:

- Wenn das ADS eine VAST-Antwort sendet, MediaTailor fügt es zu Beginn des Manifests alle Anzeigen aus der Antwort in einen Werbeblock ein. 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 jeder Werbeunterbrechung 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-Werbeblöcke erstellen möchten, Ihr ADS VMAP jedoch nicht unterstützt, stellen Sie sicher, dass das Manifest Werbemarkierungen enthält. MediaTailor fügt Werbeanzeigen an den Markierungen ein, wie in den folgenden Abschnitten beschrieben.

Verhalten beim Zusammenfügen von Live-Anzeigen

In Live-Streams führt AWS Elemental MediaTailor immer eine Werbeersetzung durch, wobei die Gesamtzeit zwischen den Werbemarkierungen möglichst exakt erhalten bleibt. Wenn Anzeigenmarkierungen das DURATION Attribut enthalten, bestimmt MediaTailor anhand des Werts die Dauer der Werbeunterbrechung. Jeder CUE-OUT-Indikator muss einen passenden CUE-IN-Indikator in Live-Workflows 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 Anzeigen

AWS Elemental MediaTailor schließt wie folgt Werbung aus der VAST-Antwort des ADS (Ad Decision Server) ein:

- Wenn eine Dauer angegeben ist, wird eine Reihe von Anzeigen MediaTailor ausgewählt, die in die Dauer passen, und diese werden aufgenommen.
- 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 während der Ersetzung von Live-Inhalten durch Werbung an die folgenden Richtlinien:

- MediaTailor versucht, komplette Anzeigen 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 als Dauer der Werbepause 70 Sekunden festgelegt sind und die ADS-Antwort zwei 40-sekündige Werbespots enthält, gibt AWS Elemental MediaTailor einen der 40-sekündigen Werbespots wieder. 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 beträgt, 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 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 Anforderung des übergeordneten HLS-Manifests anhängen `aws.insertionMode=GUIDED`.

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://bdaaeb4bd9114c088964e4063f849065.mediatailor.us-east-1.amazonaws.com
```

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

Beispiel

```
AKIAIOSFODNN7EXAMPLE
```

- `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 vorherigen Beispielen ergeben sich die folgenden vollständigen URLs.

- Beispiel:

```
https://bdaaeb4bd9114c088964e4063f849065.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.

Wird verwendet MediaTailor , um linear zusammengestellte Streams zu erstellen

AWS Elemental MediaTailorChannel 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](#)
- [Mit Programmen arbeiten](#)
- [Fügen Sie personalisierte Anzeigen und Werbeunterbrechungen in einen Channel-Stream ein](#)
- [Zeitversetzte Anzeige](#)

Mit Quellstandorten arbeiten

Ein Quellspeicherort stellt den Ursprungsserver dar, auf dem Ihr Quellinhalt gespeichert ist. Ein Quellort kann Amazon S3, ein Standard-Webserver, ein Content Delivery Network (CDN) wie Amazon CloudFront oder ein Verpackungshersteller wie sein AWS Elemental MediaPackage.

MediaTailor ruft Ihre Inhaltsmanifeste vom Quellspeicherort ab und verwendet sie, um den linearen Stream Ihres Kanals zusammenzustellen.

In diesem Thema wird erklärt, wie Sie die AWS Elemental MediaTailor Konsole verwenden, um Quellverzeichnisse zu erstellen und zu löschen, und wie Sie mit VOD Quellen arbeiten.

Themen

- [Einen Quellspeicherort erstellen](#)
- [Konfiguration der Authentifizierung für Ihren Quellstandort](#)
- [Mit VOD Quellen arbeiten](#)
- [Mit Live-Quellen arbeiten](#)
- [Verwenden von Paketkonfigurationen](#)
- [Zwischenspeichern von Manifesten](#)
- [Hinzufügen von Live-Quellen zu Ihrem Quellstandort](#)

Einen Quellspeicherort erstellen

Im folgenden Verfahren wird erklärt, wie Sie mithilfe der MediaTailor Konsole einen Quellspeicherort erstellen. Informationen zum Erstellen von Quellverzeichnissen mithilfe von finden Sie [CreateSourceLocation](#) in der AWS Elemental MediaTailor API Referenz. MediaTailor API

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 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>. Das URL muss in einem HTTP URL Standardformat mit dem Präfix <http://> oder <https://> sein.

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 Ursprung zu autorisieren. Weitere Informationen finden Sie unter [the section called “Arbeiten mit SigV4 für Amazon S3”](#).
- 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 beginn tx-amzn-. Wenn Sie eine Integration mit [MediaPackage CDNAutorisierung](#) durchführen, sollte der Header-Wert wie folgt lauten `X-MediaPackage-CDNIdentifier`.
- 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.
- ARNGeheim — Das ARN Geheimnis, 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 Base HTTP URL (die MediaTailor verwendet, um auf die Manifeste zuzugreifen) und die Standardsegmentbasis URL (die die Spieler für den Zugriff auf die Inhaltssegmente verwenden) verwenden. Wenn du keinen Wert eingibst, 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 habenURL. Bei der Basis URL kann es HTTP URL sich um einen vollständigen Pfad oder einen relativen Pfad handeln/some/path/. Die Namen werden verwendet, um zu identifizieren, welcher Server verwendet werden soll, wenn MediaTailor eine Anfrage nach Inhaltssegmenten eingeht. Wenn die Anfrage den Header enthält X-MediaTailor-SegmentDeliveryConfigurationName und der Wert des Headers mit einem Namen übereinstimmt, URL wird die entsprechende Basis für die Bereitstellung des Inhalts verwendet. 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, MediaTailor v2 und AWS Secrets Manager Zugriffstoken für die Quellstandortauthentifizierung verwendet werden.

Für weitere Informationen wählen Sie das entsprechende Thema aus.

Themen

- [Arbeiten mit SigV4 für Amazon S3](#)
- [Arbeitet mit SigV4 für Version 2 MediaPackage](#)
- [AWS Secrets Manager Arbeitet mit Zugriffstoken-Authentifizierung](#)

Arbeiten mit SigV4 für Amazon S3

Signature Version 4 (Sigv4) für Amazon S3 ist ein Signaturprotokoll, das zur Authentifizierung von Anfragen an Amazon S3 verwendet wird. HTTPS Wenn Sie SigV4 für Amazon S3 verwenden, MediaTailor fügt der HTTPS Anfrage an den Amazon S3 S3-Bucket, der als Ihr Ursprung verwendet wird, einen signierten Autorisierungsheader hinzu. 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 [Authentifizieren von Anfragen \(AWSSignature Version 4\)](#) in der Amazon S3 API S3-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 gewähren. IAM [Informationen zur Konfiguration des Zugriffs in finden Sie unter Zugriffsverwaltung im IAM 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 Quellpaketkonfigurationen verwiesen wird. VOD
- Der Aufrufer von API muss über `s3: GetObject` IAM -Berechtigungen verfügen, um alle Manifeste der obersten Ebene zu lesen, auf die in Ihren Quellpaketkonfigurationen verwiesen wird. MediaTailor VOD
- Ihre MediaTailor Ausgangsbasis URL muss dem URL Anforderungsformat 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](#).

MediaTailor SigV4-Signaturanfragen

Sie können SigV4-Signaturen für Anfragen verwenden MediaTailor, die von gültigen AWS Quellen wie Amazon S3, Channel Assembly und MediaPackage V2 gestellt wurden. Auf diese Weise können die Quellen wissen MediaTailor, von wem die Anfragen gestellt wurden, und Sie können den Zugriff auf MediaTailor Anfragen beschränken. Wenn Sie den Zugriff nicht nur auf MediaTailor Anfragen beschränken, können andere MediaTailor Kunden über ihre eigene MediaTailor Wiedergabekonfiguration auf Ihren Absender zugreifen.

Wir signieren Anfragen an folgende AWS Key Management Service Anbieter: Channel Assembly und MediaPackage V2. Der Ursprung URLs muss wie folgt aussehen:

`mediapackagev2.<region>.amazonaws.com`

`channel-assembly.mediatailor.<region>.amazonaws.com`

`s3.<region>.amazonaws.com`

Important

Verwenden Sie https, um Anfragen an Origin zu signieren URLs. Wenn Ihr Origin nicht für die Verwendung konfiguriert ist HTTPS, MediaTailor signiert Origin-Anfragen nicht mit SigV4.

IAM Beispielrichtlinien für Ursprünge, auf die der Zugriff beschränkt werden soll MediaTailor

Die folgenden IAM Richtlinien zeigen Beispiele dafür, wie der Zugriff beschränkt werden kann MediaTailor.

Amazon S3

Auf das Konto beschränkt:

```
{
  "Effect": "Allow",
```

```

    "Principal": {"Service": "mediatailor.amazonaws.com"},
    "Action": "s3:GetObject",
    "Resource": "arn:aws:s3:::mybucket/*",
    "Condition": {
      "StringEquals": {"AWS:SourceAccount": "123456789012"}
    }
  }
}

```

Auf die Wiedergabekonfiguration beschränkt: ARN

```

{
  "Effect": "Allow",
  "Principal": {
    "Service": "mediatailor.amazonaws.com"
  },
  "Action": "s3:GetObject",
  "Resource": "arn:aws:s3:::mybucket/*",
  "Condition": {
    "StringEquals": {
      "AWS:SourceArn": "arn:aws:mediatailor:us-
west-2:123456789012:playbackConfiguration/test"
    }
  }
}

```

MediaPackage V2

Auf das Konto beschränkt:

```

{
  "Effect": "Allow",
  "Principal": {
    "Service": "mediatailor.amazonaws.com"
  },
  "Action": "mediapackagev2:GetObject",
  "Resource": "arn:aws:mediapackagev2:us-
west-2:123456789012:channelGroup/emp-origin-channel-group/channel/emp-origin-channel/
originEndpoint/emp-origin-endpoint",
  "Condition": {
    "StringEquals": {
      "AWS:SourceAccount": "123456789012"
    }
  }
}

```

```
}

```

Auf die Wiedergabekonfiguration beschränkt: ARN

```
{
    "Effect": "Allow",
    "Principal": {
        "Service": "mediatailor.amazonaws.com"
    },
    "Action": "mediapackagev2:GetObject",
    "Resource": "arn:aws:mediapackagev2:us-
west-2:123456789012:channelGroup/emp-origin-channel-group/channel/emp-origin-channel/
originEndpoint/emp-origin-endpoint",
    "Condition": {
        "StringEquals": {
            "AWS:SourceArn": "arn:aws:mediatailor:us-
west-2:123456789012:playbackConfiguration/test"
        }
    }
}

```

Montage des Kanals

Auf das Konto beschränkt:

```
{
    "Effect": "Allow",
    "Principal": {
        "Service": "mediatailor.amazonaws.com"
    },
    "Action": "mediatailor:GetManifest",
    "Resource": "arn:aws:mediatailor:us-
west-2:123456789012:channel/ca-origin-channel",
    "Condition": {
        "StringEquals": {
            "AWS:SourceAccount": "123456789012"
        }
    }
}

```

Auf die Wiedergabekonfiguration beschränkt: ARN

```
{

```

```
        "Effect": "Allow",
        "Principal": {
          "Service": "mediatailor.amazonaws.com"
        },
        "Action": "mediatailor:GetManifest",
        "Resource": "arn:aws:mediatailor:us-
west-2:123456789012:channel/ca-origin-channel",
        "Condition": {
          "StringEquals": {
            "AWS:SourceArn": "arn:aws:mediatailor:us-
west-2:123456789012:playbackConfiguration/test"
          }
        }
      }
    }
  }
```

Arbeitet mit SigV4 für Version 2 MediaPackage

Signature Version 4 (Sigv4) für MediaPackage v2 ist ein Signaturprotokoll, das verwendet wird, um Anfragen an v2 zu authentifizieren. MediaPackage HTTP Wenn Sie SigV4 für MediaPackage v2 verwenden, MediaTailor wird ein signierter Autorisierungsheader in die HTTP Anfrage an den MediaPackage v2-Endpunkt aufgenommen, 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 [Authentifizieren von Anfragen \(AWSSignature Version 4\)](#) in der MediaPackage APIv2-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.
- Ihr MediaTailor Quellstandort muss ein v2-Endpunkt sein. URL MediaPackage
- Der Aufrufer von API muss über `mediapackagev2:GetObject` IAM -Berechtigungen verfügen, um alle Manifeste der obersten Ebene zu lesen, 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 Autorisierung verwenden CDN](#)
- [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 Schlüsselmaterial
- 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 Schlüssel 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 das GeheimnisARN, wodurch die Verschlüsselung auf dieses spezifische Geheimnis beschränkt wird. Als zusätzliche Sicherheitsmaßnahme gewährt es MediaTailor in Ihrem Namen einen AWS KMS Zuschuss. MediaTailor wendet eine [GrantConstraints](#) Operation an, die es uns nur ermöglicht, das mit dem im Secrets Manager ARN enthaltene Verschlüsselungskontext `SecretString` verknüpfte Geheimnis zu entschlüsseln.

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 AWS KMS im AWS Key Management Service Entwicklerhandbuch unter [Authentifizierung und Zugriffskontrolle für](#).

Um Ihren vom Kunden verwalteten Schlüssel mit Ihren MediaTailor Quellstandortressourcen zu verwenden, müssen Sie dem IAM Principal, der den Anruf durchführt, die Erlaubnis erteilen, [CreateSourceLocation](#) oder die folgenden API Operationen ausführen [UpdateSourceLocation](#) zu dürfen:

- `kms:CreateGrant`: Fügt einem kundenverwalteten Schlüssel eine Erteilung hinzu. MediaTailor gewährt Ihrem vom Kunden verwalteten Schlüssel eine Genehmigung, sodass dieser den

Schlüssel verwenden kann, um einen Quellstandort 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 an `RetireGrant`, 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 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 Autorisierung verwenden CDN”](#).

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:

- Dabei [KmsKeyId](#) muss es sich um den [Schlüssel ARN](#) des vom Kunden verwalteten Schlüssels handeln, 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 abzurufenSecretString.

Notieren Sie sich das Geheimnis 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 Autorisierung verwenden CDN

Wenn Sie Origin AWS Elemental MediaPackage als Quellstandort verwenden, MediaTailor kann die Integration mit MediaPackage Endpunkten erfolgen, die Autorisierung verwendenCDN.

Gehen Sie wie folgt vor, um die Integration mit einem MediaPackage Endpunkt durchzuführen, der CDN Autorisierung verwendet.

Zur Integration mit MediaPackage

1. Führen Sie die Schritte unter [CDNAutorisierung 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 das `KmsKeyId` mit dem vom Kunden verwalteten SchlüsselARN, den Sie in [the section called “Schritt 1: Erstellen Sie einen AWS KMS symmetrischen, vom Kunden verwalteten Schlüssel”](#).
 - (Optional) Für können Sie den `SecretString` entweder UUID auf einen neuen Wert rotieren oder Sie können das vorhandene verschlüsselte Geheimnis verwenden, sofern es sich um ein Schlüssel- und Wertepaar in einem JSON Standardformat handelt, z. `{"MediaPackageCDNIdentifier": "112233445566778899"}` B.
4. Führen Sie die Schritte unter [the section called “Anhängen einer ressourcenbasierten Geheimrichtlinie”](#).
 5. Führen Sie die Schritte unter [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 Herkunftsaauthentifizierung 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 das Geheimnis ARN in Ihre Zugriffskonfiguration für den Quellspeicherort auf.
2. MediaTailor erstellt eine Genehmigung für den vom Kunden verwalteten Schlüssel, sodass dieser den Schlüssel verwenden MediaTailor kann, um auf das in der gespeicherte [Zugriffstoken](#) zuzugreifen und es zu entschlüsseln. 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 [RevokeGrant](#)in der AWS Key Management Service APIReferenz.

3. Wenn eine VOD Quelle erstellt, aktualisiert oder in einem Programm verwendet wird, werden HTTP Anfragen an die Quellverzeichnisse gestellt, MediaTailor um die mit den VOD Quellen am Quellspeicherort verknüpften Quellinhaltsmanifeste abzurufen. Wenn die VOD Quelle einem

Quellspeicherort 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 Quellort 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 Speicherort für das Manifest 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 verpackt ist — HLS und DASH —, dann 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 Werbemarke in allen Paketkonfigurationen mit demselben Offset erscheinen und eine Dauer von Null haben. Denn HLS, MediaTailor wird erkannt DATERANGE und EXT-X-CUE-OUT markiert. Für DASH, HLS erkennt das erste Event-Tag in jedem EventStream Tag.

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 Quellort

Das folgende Verfahren erklärt, wie Sie VOD Quellen zu Ihrem Quellspeicherort hinzufügen und Paketkonfigurationen mithilfe der MediaTailor Konsole einrichten. Informationen zum Hinzufügen von VOD Quellen mithilfe von finden Sie [CreateVodSource](#) in der AWS Elemental MediaTailor API Referenz. MediaTailor API

Important

Bevor Sie Ihre VOD Quellen hinzufügen, stellen Sie sicher, dass sie die folgenden Anforderungen erfüllen:

- 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 bieten wir keine Unterstützung pro Titel oder automatisiert an 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 Quellorten 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 [Um einen Quellspeicherort zu erstellen](#) Verfahren erstellt haben.
4. Wählen Sie VODQuelle hinzufügen aus.
5. Geben Sie unter VODQuelldetails einen Namen für Ihre VOD Quelle ein:
 - Name: Eine Kennung für Ihre VOD Quelle, z. my-example-videoB.
6. Unter Paketkonfigurationen > *source-group-name* geben Sie 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 [Verwenden Sie Quellgruppen mit den Ausgängen Ihres Kanals](#).
- Typ: Wählen Sie das Paketformat für diese Konfiguration aus. MediaTailor unterstützt HLS und DASH.
- Relativer Pfad: Der relative Pfad von der Base des Quellverzeichnisses HTTP URL 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

Quellversionen (DASH) oder Variantenstreams () separate Paketkonfigurationen erstellen. HLS

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.

Mit Live-Quellen arbeiten

Eine Live-Quelle steht für einen einzelnen Live-Stream, z. B. ein Fußball-Live-Spiel 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:

- VODQuellen für einen Kanal, der VOD Live-Inhalte enthält
- Live-Quellen für einen Kanal, der live-to-live Inhalte enthält, die mit VOD Live-Inhalten vermischt sind

Ein Beispiel für VOD Live-Inhalte ist ein Kanal, der eine Bibliothek von VOD Assets zu einem Live-Stream zusammenfügt. Ein Beispiel für live-to-live Inhalte, die mit VOD 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 Live-Inhalten gemischt werden, ist ein live-to-live reiner 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 Programmübersreibungen beinhaltet und Inhalte enthältVOD. Dazu verwenden Sie ein Encoder/Packager-Paar für die nationalen Inhalte und dann regionale Encoder, wenn diese Regionen live sind. Anschließend erstellen Sie regionale Channel-Assembly Channels mit jeweils eigenen Zeitplänen. Auf diese Weise können Zuschauer nach

Bedarf hin und her wechseln. Dieses Setup hilft Ihnen, die Kosten für Codierungs-/Paketierung zu minimieren.

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 verpackt ist — HLS und DASH —, dann 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 `DATERANGE` konfigurieren.
- **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 | Endpointtyp | Apple HLS | Erforderlich, sofern nicht verwendet CMAF | Um HLS ts AWS Elemental MediaConvert Jobs zuzuordnen |
| HLS | Endpointtyp | CMAF | Erforderlich, sofern Sie Apple nicht verwenden HLS | Um HLS mp4 AWS Elemental MediaConvert Jobs zuzuordnen |
| 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 Werbermarkierungen | |

| Standard | Einstellung | Wert | Notwendigkeit | Hinweise |
|----------|-------------------------|---|--|--|
| HLS | IncludeIframeOnlyStream | Disabled | Empfohlen | |
| DASH | ManifestLayout | FULL | Empfohlen | |
| 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 von finden Sie [CreateLiveSource](#) in der AWS Elemental MediaTailor API Referenz. MediaTailor API

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 Quellverzeichnisse den Quellspeicherort aus, den Sie in dem [Um einen Quellspeicherort zu erstellen](#) 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. Unter Paketkonfigurationen > *source-group-name* geben Sie Informationen zur Paketkonfiguration ein:

Note

Innerhalb einer Paketkonfiguration müssen alle VOD Quellen und Live-Quellen dieselbe Anzahl von 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 Sie Quellgruppen mit den Ausgängen Ihres Kanals](#).
- Typ: Wählen Sie das Paketformat für diese Konfiguration aus. MediaTailor unterstützt HLS und DASH.
- Relativer Pfad: Der relative Pfad von der Base des Quellverzeichnisses HTTP URL 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 Quellversionen (DASH) oder Variantenstreams () separate Paketkonfigurationen erstellen. HLS

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.

Hinzufügen von Live-Quellen zu Ihrem Quellstandort

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 von finden Sie [CreateLiveSource](#) in der AWS Elemental MediaTailor APIReferenz. MediaTailor API

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 Quellverzeichnisse den Quellspeicherort aus, den Sie in dem [Um einen Quellspeicherort zu erstellen](#) 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. Unter Paketkonfigurationen > *source-group-name* geben Sie Informationen zur Paketkonfiguration ein:

Note

Innerhalb einer Paketkonfiguration müssen alle VOD Quellen und Live-Quellen dieselbe Anzahl von 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 Sie Quellgruppen mit den Ausgängen Ihres Kanals](#).
- Typ: Wählen Sie das Paketformat für diese Konfiguration aus. MediaTailor unterstützt HLS und DASH.
- Relativer Pfad: Der relative Pfad von der Base des Quellverzeichnis HTTP URL 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 Quellversionen (DASH) oder Variantenstreams () separate Paketkonfigurationen erstellen. HLS

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.

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 einer VOD Quelle oder einer Live-Quelle zugeordnet.

Themen

- [Kanäle erstellen](#)

- [Verwenden Sie Quellgruppen mit den Ausgängen Ihres Kanals](#)
- [Kanäle löschen](#)

Kanäle erstellen

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. Gib 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 verwendet `programRules` , wenn Sie sie `audienceMedia` für Ihr Standardprogramm erstellen.
- >Wählen Sie Hinzufügen.
 - 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 VODQuellnamen, der 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 Zielgruppendetails in den Programmregeln an.

8. Wenn Sie in den Kanaldetails die Option Standard auswählen, können Sie die Zielgruppen unter Zielgruppendetails definieren. Diese Zielgruppen werden verwendet, `programRules` wenn du sie `audienceMedia` für dein Standardprogramm erstellst:

- 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 Quellort](#).

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 in ungültige Programme mit EXT-X-DATERANGE Tags MediaTailor einzufügen. Wählen Sie Scte35 Enhanced diese Option aus, um Werbeunterbrechungen mithilfe von EXT-X-CUE-OUT EXT-X-CUE-IN UND-Tags in VOD Programme MediaTailor einzufügen. 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. Wähle unter Kanalrichtlinie die IAM Richtlinieneinstellungen deines Kanals aus:
 - Füge keine Kanalrichtlinie hinzu: Beschränke die Wiedergabe nur auf Personen, die Zugriff auf die Zugangsdaten dieses Accounts 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. Ihr Kanal ist erst aktiv, wenn Sie ihn mit der MediaTailor Konsole oder dem starten MediaTailor StartChannel API.

Verwenden Sie Quellgruppen mit den Ausgängen Ihres 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.

Beispielsweise:

- VOD-Quellen 1 und 2 haben beide drei Paketkonfigurationen mit den Quellgruppen: HLS-DASH, und HLS-4k.
- VOD-Quelle 3 hat zwei Paketkonfigurationen mit Quellgruppen HLS und DASH.

Wenn Kanal A zwei Ausgänge mit 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 Quellgruppenbezeichnungen HLS und DASH haben.

Wenn Kanal B zwei Ausgänge mit Quellgruppen HLS und HLS-4k hat, kann er VOD-Quelle 1 und 2 verwenden, aber nicht 3. Das liegt daran, dass die VOD-Quellen 1 und 2 beide Paketkonfigurationen mit 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.

Kanäle löschen

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.

Mit Programmen arbeiten

Jedes Programm enthält eine VOD Quelle oder eine Live-Quelle, die Teil eines Quellspeicherorts 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 eine Slate, bei der 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

- [Programme erstellen](#)
- [Mit Programmregeln arbeiten](#)
- [Zielgruppen definieren](#)

Programme erstellen

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 mit dem MediaTailor API finden Sie [CreateProgram](#) in der AWS Elemental MediaTailor APIReferenz.

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.
- VODQuellname: Der Name der VOD Quelle, die mit dem Programm MediaTailor verknüpft ist:
 - Wenn Sie Select a existing VOD source wählen, wählen Sie einen VOD Quellnamen aus der Liste der VOD Quellen aus, die mit Ihrem Konto verknüpft 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.
- VODQuellname: Der Name der VOD Quelle, die mit dem Programm MediaTailor verknüpft 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 Ihres 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 weitergeleitet.

7. Wählen Sie Weiter aus, um zu Werbeunterbrechungen hinzufügen zu wechseln.
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.
 - VODQuellname: Wählen Sie „Vorhandene VOD Quelle auswählen“ und wählen Sie die VOD Quelle aus, die Sie für Slate verwenden, 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 Programmquelle (alle Audio-, Video- und Untertitelspuren) ausgerichtet ist. Andernfalls wird die Werbeunterbrechung übersprungen. Wenn Sie beispielsweise 0 eingeben, wird eine Pre-Roll-Werbeunterbrechung erstellt, die vor Programmbeginn abgespielt wird.
 - Verfügbare Zahl: MediaTailor [schreibt, in welche dieser Wert geschrieben wirdsplice_insert.avail_num, wie in Abschnitt 9.7.3.1 der Spezifikation SCTE -35, Meldung zum Einsetzen eines digitalen Programms, 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 -35-Spezifikation definiert. SCTE Der Standardwert lautet 0. Die Werte müssen zwischen 0 und 256 (einschließlich) liegen.
 - ID des Splice-Ereignisses: MediaTailor schreibt diesen Wert `splice_insert.splice_event_id`, wie in Abschnitt 9.7.3.1 der -35-Spezifikation definiert. SCTE Der Standardwert lautet 1.
 - Eindeutige Programm-ID: MediaTailor schreibt diesen Wert `splice_insert.unique_program_id`, wie in Abschnitt 9.7.3.1 der -35-Spezifikation

definiert. SCTE 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 „Einen vorhandenen Quellspeicherort auswählen“ wählen, wählen Sie einen Quellspeicherort aus dem Drop-down-Menü „Quellspeicherort auswählen“ 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.

- VODQuellname: Der Name der VOD Quelle, die dem Programm zugeordnet werden soll.

Wenn Sie Select a existing VOD source wählen, wählen Sie einen VOD Quellennamen aus der Liste der VOD Quellen aus, die mit Ihrem Konto verknüpft 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 weitergeleitet.

14. Wählen Sie Werbeunterbrechung hinzufügen aus. Konfigurieren Sie unter Werbeunterbrechungen die Einstellungen für die Werbeunterbrechung:

- Name des Slate-Quellspeicherorts: Wählen Sie „Bestehenden Quellspeicherort auswählen“ und wählen Sie den Quellspeicherort aus, an dem Ihr Slate gespeichert ist, den Sie zuvor in diesem Tutorial erstellt haben.
- VODQuellname: Wählen Sie „Eine vorhandene VOD Quelle auswählen“ und wählen Sie die VOD Quelle aus, die Sie für Slate verwenden, 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 Programmquelle (alle Audio-, Video- und Untertitelspuren) ausgerichtet ist. Andernfalls wird die Werbeunterbrechung übersprungen. Wenn Sie beispielsweise 0 eingeben, wird eine Pre-Roll-Werbeunterbrechung erstellt, die vor Programmbeginn abgespielt wird.

 Note

Wenn in Ihrer VOD Quelle Werbemarken wie DATERANGE oder EXT-X-CUE-OUT für HLS und EventStream für mit einer Dauer von Null MediaTailor erkannt werden DASH, können Sie den Offset dieser Werbemarken aus dem Drop-down-Menü auswählen, der als Offset für die Werbeunterbrechung verwendet werden soll. Damit eine Werbechance erkannt werden kann, 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 dieses `splice_insert.avail_num`, wie in Abschnitt 9.7.3.1 der -35-Spezifikation definiert, geschrieben. SCTE Der Standardwert ist 0. Die Werte müssen zwischen und (einschließlich) liegen. 0 256

Für Avail Expected wird dieses `splice_insert.avails_expected`, wie in Abschnitt 9.7.3.1 der -35-Spezifikation definiert, geschrieben. SCTE Der Standardwert ist 0. Die Werte müssen zwischen und (einschließlich) liegen. 0 256

Für die Splice-Event-ID wird in diese geschriebensplice_insert.splice_event_id, wie in Abschnitt 9.7.3.1 der -35-Spezifikation definiert. SCTE Der Standardwert ist 1.

Bei der eindeutigen Programm-ID wird in diese geschriebensplice_insert.unique_program_id, wie in Abschnitt 9.7.3.1 der -35-Spezifikation definiert. SCTE Der Standardwert ist 0. Die Werte müssen zwischen 0 und (einschließlich) liegen. 256

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.

Mit Programmregeln arbeiten

WithProgram Regeln: Sie können 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 STANDARD Stufenkanälen mit LINEAR Wiedergabemodus 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 tun MediaTailor `CreateChannelAPI`. Jede Zielgruppe muss zwischen 1 und 32 alphanumerische Zeichen lang sein. Wenn die für die Zielgruppen angegebenen Werte ungültig sind, schlägt die Anfrage fehl.

Im LINEAR Wiedergabemodus können Sie nur Zielgruppen auf STANDARD Tier-Kanälen definieren.

Wenn Sie die Zielgruppen aktualisieren müssen, können Sie dies entweder über die MediaTailor Konsole oder die tun MediaTailor `UpdateChannelAPI`.

Wenn Sie die `ProgramRules` Funktion verwenden, stellen Sie sicher, dass die in `CreateProgram` oder in der `UpdateProgram` Anfrage `AudienceMedia` definierte Zielgruppe die bestehende Zielgruppe enthält, die im Kanal definiert ist.

Alternative Medien erstellen

In der folgenden Aufgabe wird erklärt, wie alternative Medien mithilfe der MediaTailor Konsole definiert werden. Informationen zur Definition alternativer Medien mithilfe von finden Sie [CreateProgram](#) in der AWS Elemental MediaTailor APIReferenz. MediaTailor API

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 Channel-Assembly > Channels aus.
3. Wählen Sie den Kanalnamen aus, dem Sie alternative Medien hinzufügen möchten.
4. Erstellen Sie ein Programm. Weitere Informationen finden Sie unter [Programme erstellen](#).
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 VOD Sie als Quelltyp aus.
- (Optional) Geben Sie einen Clip-Bereich an. Bei VOD Quellen, einschließlich VOD Quellen für alternative Medien, können Sie angeben, ob ein Teil einer VOD Quelle abgespielt werden soll, wobei der Ausschnitt am Anfang und/oder am Ende der Quelle ausgeschnitten wird. Geben Sie an. Die Anfangs- und Endversätze werden in Millisekunden angegeben.
- (Optional) a\ Werbeunterbrechungen hinzufügen. Dies erfolgt auf die gleiche Weise wie beim Erstellen von Programmen. Weitere Informationen finden Sie unter [Programme erstellen](#).

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 [Programme erstellen](#).

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 von Ihnen dynamisch angehängt CDN oder über einen Aufruf zu Ihrem Content- oder Kundenverwaltungssystem hinzugefügt werden. Sie müssen die Zuordnung einer bestimmten Playback-Sitzung zu einer `audienceId` externen Sitzung aufrechterhalten. 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 Kanalzusammenstellung monetarisieren MediaTailor, indem du Werbeunterbrechungen in deine Programme einfügst, ohne den Inhalt mit SCTE -35 Markierungen

zu versehen. Sie können Channel Assembly mit dem Dienst zur MediaTailor Anzeigeneinfügung oder mit jeder beliebigen serverseitigen Anzeigeneinfügung () verwenden. SSAI

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, URL ist der Endpunkt 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 DASH Land-Streams für die MediaTailor Anzeigeneinblendung 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 Entscheidungsserver für Anzeigen (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 System so zu konfigurieren, dass es die Manifeste aus der Channel-Assembly als Proxy weiterleitet und dann das CDN Präfix URLs an der Inhaltsquelle URL verwendet.

MediaTailor Für das Einfügen von Anzeigen konfigurieren

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: Geben Sie die Wiedergabe URL vom Ausgang Ihres Kanals ein, abzüglich des Dateinamens und der Erweiterung. Weiterführende Informationen zur MediaTailor Konfiguration finden Sie unter [Erforderliche Einstellungen](#).
 - Entscheidungsserver hinzufügen: Geben Sie den URL für Ihren einADS.
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 der MediaTailor Anzeigeneinfügung](#).

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. Du verwendest Nachrichten, die auf der SCTE -35-Spezifikation basieren, um den Inhalt auf Werbeunterbrechungen vorzubereiten. Beispielsweise können Sie Nachrichten mit dem Wert SCTE -35 verwenden, um Metadaten zu den Werbeunterbrechungen bereitzustellen. Weitere Informationen zur SCTE -35-Spezifikation finden Sie unter [Cueing-Meldung beim Einfügen digitaler Programme](#).

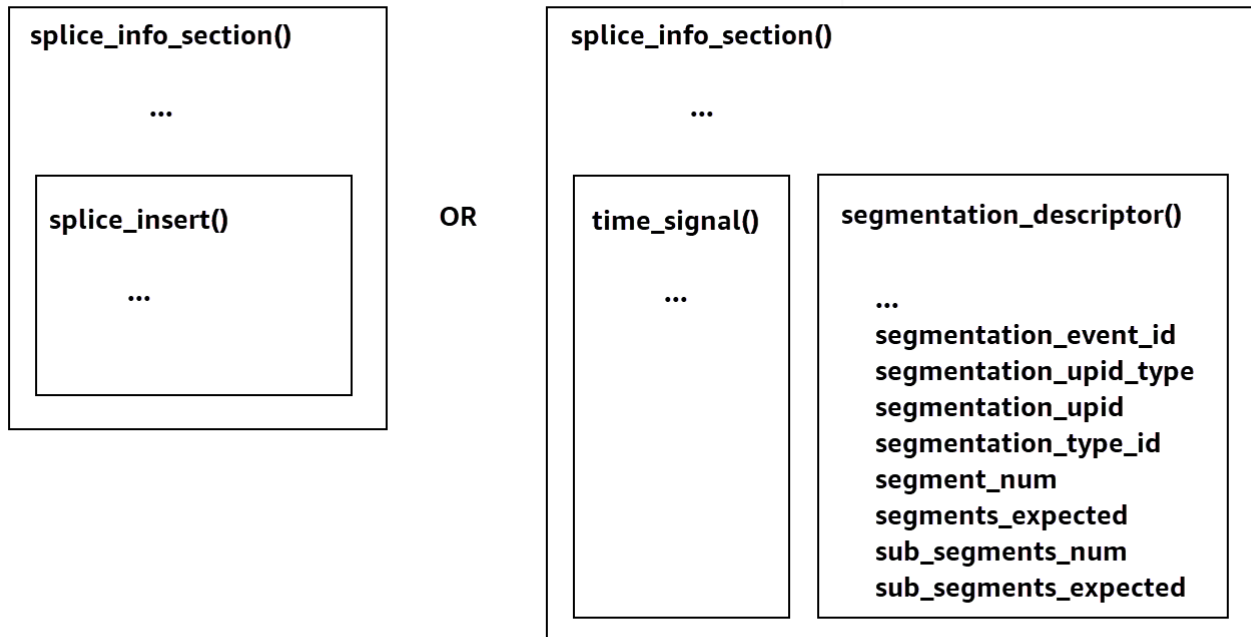
Sie können die Werbeunterbrechungen auf eine von zwei Arten einrichten:

- Eine `time_signal` SCTE -35-Nachricht mit einer `segmentation_descriptor` Nachricht anhängen. 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 Anzeigenmarkierung EXT-X-DATERANGE (HLS) oder EventStream (DASH) in das Ausgabemanifest.
- Anfügen 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 im 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 EXT-X-CUE-OUT und enthält die Metadaten für Werbeunterbrechungen, die AdBreak bei der Erstellung oder Aktualisierung des Programms angegeben wurden. Es enthält CAID immer.
 - `EXT-0ATCLS-SCTE35`: Dieses Tag erscheint im selben Segment wie die Nachricht EXT-X-CUE-OUT -35 und enthält Base64-kodierte Byte. SCTE
 - `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 Werbeunterbrechung vorbei ist, und weist auf einen Schnitt von einer Werbepause zurück zum Inhalt hin.

Die folgende Abbildung zeigt die beiden Möglichkeiten, Werbeunterbrechungen in einem Kanal mit SCTE -35 Nachrichten einzurichten:

- Verwende 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 Spezifikation 2022 SCTE -35, Meldung zum [Einfügen von digitalen Programmen](#).

Die Informationen zum Werbeunterbruch werden in den `splice_info_section` SCTE Ausgabedaten -35 angezeigt. Mit MediaTailor können Sie eine einzelne `segmentation_descriptor` Nachricht mit einer einzelnen `time_signal` Nachricht verbinden.

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 Spezifikation 2022 SCTE -35, die Sie auf der ANSI Webstore-Website erwerben können.](#)

Pflichtfelder für eine Nachricht `segmentation_descriptor`

| 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 an <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> . Der Wert muss eine hexadezimale Zeichenfolge sein, die die Zeichen 0-9 und enthält. A-F |

| Feld | Typ | Standardwert | Beschreibung |
|----------------------|----------|--------------|---|
| 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 |

Sie haben gelernt, wie Sie beim Kanalaufbau Werbeunterbrechungen mithilfe von SCTE -35 Nachrichten einrichten, die Struktur und die erforderlichen Felder für diese Nachrichten sowie Muster HLS und DASH Ausgaben, die die SCTE -35-Nachrichten enthalten, kennen gelernt.

Zeitversetzte Anzeige

Zeitversetztes Ansehen bedeutet, dass Zuschauer einen Livestream zu einem früheren Zeitpunkt als dem Zeitpunkt der Anfrage ansehen können, sodass sie von Anfang an an einem laufenden Programm teilnehmen oder sich ein bereits abgeschlossenes Programm ansehen können. 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 im V1-Benutzerhandbuch unter [Referenz zum zeitversetzten Anzeigen. 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](#)
- [Verwendung von zeitversetzter Anzeige mit CDNs](#)

Zeitversetzende Parameter für Manifestanfragen

In diesem Abschnitt sind die Parameter für zeitversetzende Manifestanfragen aufgeführt.

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
```

Start time (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`
- POSIX-Zeit (oder Epochenzeit), 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. MediaTailor Wählt andernfalls eine Verzögerung, 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.mediataylor.us-west-2.amazonaws.com/v1/channel/  
ExampleChannel/index_dash.mpd?aws.mediataylor.channel.startProgram=SuperBowlLVII
```

Verwendung von zeitversetzter Anzeige mit CDNs

Um bei Angabe einer Startzeit oder eines Startprogramms ein gleitendes Fenster zu erreichen, wird die Startzeit MediaTailor oder der Wert für das Startprogramm in eine entsprechende Zeitverzögerung umgerechnet. 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.

HLS-Beispiel

Angenommen, Sie fordern mithilfe einer URL wie der folgenden ein primäres HLS-Manifest mit einer Startzeit an:

```
https://<some prefix>.channel-assembly.mediataylor.us-west-2.amazonaws.com/v1/channel/  
ExampleChannel/  
index_hls.m3u8?aws.mediataylor.channel.startTime=2017-12-19T13:00:28-08:00
```

MediaTailor antwortet mit einem Manifest, das Zeitverzögerungsparameter für die URLs des untergeordneten Manifests enthält. Wenn Sie das Manifest beispielsweise zu einem Zeitpunkt anfordern `2017-12-19T13:20:28-08:00`, also 1200 Sekunden nach der angeforderten Startzeit, MediaTailor antwortet 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.mediataylor.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.mediataylor.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.mediataylor.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.mediataylor.us-west-2.amazonaws.com/v1/channel/  
ExampleChannel/  
index_dash.mpd?aws.mediataylor.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.

Sicherheit in AWS Elemental MediaTailor

Cloud-Sicherheit AWS hat 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](#)
- [Überprüfung der Einhaltung von Vorschriften für AWS Elemental MediaTailor](#)
- [Resilienz in AWS Elemental MediaTailor](#)
- [Sicherheit der Infrastruktur in AWS Elemental MediaTailor](#)
- [Serviceübergreifende Confused-Deputy-Prävention](#)
- [Einloggen und Überwachen AWS Elemental 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 im [Abschnitt Datenschutz FAQ](#). Informationen zum Datenschutz in Europa finden Sie im [AWS Shared Responsibility Model und](#) im GDPR Blogbeitrag im AWS Security Blog.

Aus Datenschutzgründen empfehlen wir, dass Sie Ihre 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 eine Multi-Faktor-Authentifizierung (MFA).
- Verwenden Sie SSL/TLS, um mit AWS Ressourcen zu kommunizieren. Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Einrichtung API und Protokollierung von Benutzeraktivitäten mit AWS CloudTrail.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen darin enthaltenen Standardsicherheitskontrollen AWS -Services.
- Verwenden Sie erweiterte verwaltete Sicherheitservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu schützen.
- Wenn Sie FIPS 140-3 validierte kryptografische Module für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine benötigen API, verwenden Sie einen Endpunkt. FIPS Weitere Informationen zu den verfügbaren FIPS Endpunkten finden Sie unter [Federal Information Processing Standard](#) () 140-3. FIPS

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 arbeiten oder sie anderweitig AWS -Services verwenden, API, AWS CLI oder. 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 einem externen Server eine URL zur Verfügung stellen, empfehlen wir dringend, dass Sie keine Anmeldeinformationen in den angeben URL, um Ihre Anfrage an diesen Server zu überprüfen.

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 jegliche Verwendung von AWS Elemental MediaTailor, einschließlich der Konsole, APISDKs, und der AWS Command Line Interface (AWS CLI). Alle Daten, die Sie in den Service eingeben, werden möglicherweise in Diagnoseprotokolle aufgenommen.

Wenn Sie einem externen Server eine URL zur Verfügung stellen, geben Sie keine unverschlüsselten Anmeldeinformationen an, URL um Ihre Anfrage an diesen Server zu überprüfen.

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. IAMAdministratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um MediaTailor Ressourcen zu verwenden. IAM ist eine AWS -Service , die Sie ohne zusätzliche Kosten verwenden 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

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, in der Sie arbeiten 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 anzufordern müssen. Wenn Sie in nicht auf eine Funktion zugreifen können MediaTailor, finden Sie weitere Informationen unter [Fehlerbehebung bei AWS Elemental MediaTailor Identität und Zugriff](#).

Serviceadministrator — Wenn Sie in Ihrem Unternehmen für 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 Anfragen an Ihren IAM Administrator senden, um die Berechtigungen Ihrer Servicebenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die grundlegenden Konzepte von zu verstehen IAM. Weitere Informationen darüber, wie Ihr Unternehmen IAM mit verwenden kann MediaTailor, finden Sie unter [Wie AWS Elemental MediaTailor funktioniert mit IAM](#).

IAM Administrator — Wenn Sie ein IAM Administrator sind, möchten Sie vielleicht mehr darüber erfahren, wie Sie Richtlinien schreiben können, um den Zugriff darauf zu verwalten MediaTailor. Beispiele für MediaTailor identitätsbasierte Richtlinien, die Sie in verwenden können IAM, 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 übernehmen. 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-) Nutzer, die Single-Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie sich als föderierte Identität

anmelden, hat Ihr Administrator zuvor einen Identitätsverbund mithilfe von Rollen eingerichtet. IAM Wenn Sie AWS mithilfe eines Verbunds darauf zugreifen, 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, mit der Sie Ihre Anfragen mithilfe Ihrer Anmeldeinformationen kryptografisch signieren können. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode, um Anfragen selbst zu [signieren, finden Sie im IAMBenutzerhandbuch unter AWS API Anfragen signieren](#).

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen angeben. 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 [Verwenden der Multi-Faktor-Authentifizierung \(MFA\) AWS im IAM Benutzerhandbuch](#).

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, 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 und verwenden Sie diese, 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 im Benutzerhandbuch unter [Aufgaben, für die Root-Benutzeranmeldedaten erforderlich](#) sind. IAM

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 [IAMBenutzer](#) ist eine Identität innerhalb Ihres Unternehmens AWS-Konto , die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. Wir empfehlen, sich nach Möglichkeit auf temporäre Anmeldeinformationen zu verlassen, anstatt IAM Benutzer mit langfristigen Anmeldeinformationen wie Passwörtern und Zugriffsschlüsseln zu erstellen. Wenn Sie jedoch spezielle Anwendungsfälle haben, für die langfristige Anmeldeinformationen von IAM Benutzern erforderlich sind, empfehlen wir, die Zugriffsschlüssel abwechselnd zu verwenden. Weitere Informationen finden Sie im Benutzerhandbuch unter [Regelmäßiges Rotieren von Zugriffsschlüsseln für Anwendungsfälle, für die IAM langfristige Anmeldeinformationen erforderlich](#) sind.

Eine [IAMGruppe](#) 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 [Wann sollte ein IAM Benutzer \(statt einer Rolle\) erstellt werden?](#) im IAMBenutzerhandbuch.

IAMRollen

Eine [IAMRolle](#) ist eine Identität innerhalb von Ihnen AWS-Konto , für die bestimmte Berechtigungen gelten. Sie ähnelt einem IAM Benutzer, ist jedoch keiner bestimmten Person zugeordnet. Sie können vorübergehend eine IAM Rolle in der übernehmen, AWS Management Console indem Sie die [Rollen](#)

[wechseln](#). Sie können eine Rolle übernehmen, indem Sie eine AWS CLI AWS API OR-Operation aufrufen oder eine benutzerdefinierte Operation verwenden URL. Weitere Informationen zu Methoden zur Verwendung von Rollen finden Sie [unter Verwenden von IAM Rollen](#) im IAM Benutzerhandbuch.

IAM Rollen mit temporären Anmeldeinformationen sind in den folgenden Situationen nützlich:

- **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 im IAM Benutzerhandbuch unter [Erstellen einer Rolle für einen externen Identitätsanbieter](#). Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Um zu kontrollieren, 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 verschiedene Berechtigungen für eine bestimmte Aufgabe zu übernehmen.
- **Kontoübergreifender Zugriff** — Sie können eine IAM Rolle verwenden, um jemandem (einem vertrauenswürdigen Principal) 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 zum Unterschied zwischen Rollen und ressourcenbasierten Richtlinien für den kontenübergreifenden Zugriff finden Sie [IAM im Benutzerhandbuch unter Kontoübergreifender Ressourcenzugriff](#). IAM
- **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.
- **Zugriffssitzungen weiterleiten (FAS)** — Wenn Sie einen IAM Benutzer oder eine 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 an aufruft AWS -Service, kombiniert mit der Anforderung, Anfragen AWS -Service an 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 beim Stellen von FAS Anfragen finden Sie unter [Zugriffssitzungen weiterleiten](#).

- **Servicerolle** — Eine Servicerolle ist eine [IAMRolle](#), die ein Dienst übernimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM Administrator kann eine Servicerolle von innen heraus erstellen, ändern und löschenIAM. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter [Erstellen einer Rolle zum Delegieren von Berechtigungen AWS -Service an eine](#).
- **Dienstbezogene Rolle** — Eine dienstverknüpfte 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 dienstbezogene 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 IAMBenutzerhandbuch unter [Verwenden einer IAM Rolle zur Erteilung von Berechtigungen für Anwendungen, die auf EC2 Amazon-Instances ausgeführt werden](#).

Informationen darüber, ob Sie IAM Rollen oder IAM Benutzer verwenden sollten, finden [Sie im Benutzerhandbuch unter Wann sollte eine IAM Rolle \(anstelle eines Benutzers\) erstellt werden?](#). IAM

Verwalten des Zugriffs mit Richtlinien

Sie steuern 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. Berechtigungen in den Richtlinien bestimmen, ob die Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden in AWS Form von JSON Dokumenten gespeichert. Weitere Informationen zur Struktur und zum Inhalt von JSON Richtliniendokumenten finden Sie im IAMBenutzerhandbuch unter [Überblick über JSON Richtlinien](#).

Administratoren können mithilfe von AWS JSON Richtlinien festlegen, wer Zugriff auf was hat. Das bedeutet, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Um Benutzern die Erlaubnis zu erteilen, Aktionen mit den Ressourcen durchzuführen, die sie benötigen, kann ein IAM Administrator IAM Richtlinien erstellen. Der Administrator kann dann die IAM Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen übernehmen.

IAM Richtlinien definieren Berechtigungen für eine Aktion, unabhängig von der Methode, mit der Sie den Vorgang ausführen. Angenommen, es gibt eine Richtlinie, die Berechtigungen für die `iam:GetRole`-Aktion erteilt. Ein Benutzer mit dieser Richtlinie kann Rolleninformationen aus dem AWS Management Console AWS CLI, dem oder dem abrufen AWS API.

Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind Dokumente mit JSON Berechtigungsrichtlinien, die Sie an eine Identität anhängen können, z. B. an einen IAM Benutzer, eine Benutzergruppe oder eine Rolle. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen einer identitätsbasierten Richtlinie finden Sie unter [IAM Richtlinien erstellen im Benutzerhandbuch](#). IAM

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 zur Auswahl zwischen einer verwalteten Richtlinie und einer Inline-Richtlinie finden Sie im IAM Benutzerhandbuch unter [Auswahl zwischen verwalteten Richtlinien und Inline-Richtlinien](#).

Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON Richtliniendokumente, die Sie an eine Ressource anhängen. Beispiele für ressourcenbasierte Richtlinien sind IAM Rollenvertrauensrichtlinien und Amazon S3 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 nicht IAM 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 AWS WAF Amazon VPC sind Beispiele für Dienste, die Unterstützung bieten ACLs. 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 eine erweiterte Funktion, mit der Sie die maximalen Berechtigungen festlegen, die eine identitätsbasierte Richtlinie einer IAM Entität (IAM Benutzer oder Rolle) gewähren 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 zu Berechtigungsgrenzen finden Sie im IAM Benutzerhandbuch unter [Berechtigungsgrenzen für IAM Entitäten](#).
- **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 AWS-Konten Unternehmenseigentümer. 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. Weitere Informationen zu

Organizations und SCPs finden Sie unter [Richtlinien zur Servicesteuerung](#) 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 im IAMBenutzerhandbuch unter [Sitzungsrichtlinien](#).

Mehrere Richtlinientypen

Wenn mehrere auf eine Anforderung mehrere Richtlinientypen angewendet werden können, sind die entsprechenden Berechtigungen komplizierter. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie im IAMBenutzerhandbuch unter [Bewertungslogik für Richtlinien](#).

Wie AWS Elemental MediaTailor funktioniert mit IAM

Informieren Sie sich vor der Verwendung IAM zur Verwaltung des Zugriffs auf MediaTailor, welche IAM Funktionen zur Verwendung verfügbar sind MediaTailor.

IAMFunktionen, die Sie zusammen verwenden können AWS Elemental MediaTailor

| IAMMerkmal | MediaTailor Unterstützung |
|--|---------------------------|
| Identitätsbasierte Richtlinien | Ja |
| Ressourcenbasierte Richtlinien | Ja |
| Richtlinienaktionen | Ja |
| Richtlinienressourcen | Nein |
| Richtlinienbedingungsschlüssel (servicespezifisch) | Ja |
| ACLs | Nein |

| IAMMerkmal | MediaTailor Unterstützung |
|--|---------------------------|
| ABAC(Tags in Richtlinien) | Teilweise |
| Temporäre Anmeldeinformationen | Ja |
| Hauptberechtigungen | Ja |
| Servicerollen | Nein |
| Serviceverknüpfte Rollen | Ja |

Einen allgemeinen Überblick darüber, wie MediaTailor und wie andere AWS Dienste mit den meisten IAM Funktionen funktionieren, finden Sie IAM im IAMBenutzerhandbuch unter [AWS Dienste, die mit funktionieren](#).

Identitätsbasierte Richtlinien für MediaTailor

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identitätsbasierte Richtlinien sind Dokumente mit JSON Berechtigungsrichtlinien, die Sie an eine Identität anhängen können, z. B. an einen IAM Benutzer, eine Benutzergruppe oder eine Rolle. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen einer identitätsbasierten Richtlinie finden Sie unter [IAMRichtlinien erstellen im Benutzerhandbuch](#). IAM

Mit IAM identitätsbasierten Richtlinien können Sie zulässige oder verweigerte Aktionen und Ressourcen sowie die Bedingungen angeben, unter denen Aktionen zulässig oder verweigert werden. Sie können den Prinzipal nicht in einer identitätsbasierten Richtlinie angeben, da er für den Benutzer oder die Rolle gilt, dem er zugeordnet ist. Weitere Informationen zu allen Elementen, die Sie in einer JSON Richtlinie verwenden können, finden Sie im IAMBenutzerhandbuch unter [Referenz zu IAM JSON Richtlinienelementen](#).

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 an einen Kanal angehängt ist. Diese Richtlinie definiert, welche Principals Aktionen auf dem Kanal ausführen können.

Ressourcenbasierte Richtlinien sind JSON Richtliniendokumente, die Sie an eine Ressource anhängen. Beispiele für ressourcenbasierte Richtlinien sind IAM Rollenvertrauensrichtlinien und Amazon S3 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 den kontoübergreifenden Zugriff zu ermöglichen, können Sie in einer ressourcenbasierten Richtlinie ein ganzes Konto oder IAM Entitäten in einem anderen Konto als Prinzipal 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 gewähren. 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 [IAMim IAMBenutzerhandbuch unter Kontoübergreifender Ressourcenzugriff](#).

Informationen zum Anhängen einer ressourcenbasierten Richtlinie an einen Channel finden Sie unter [Kanäle erstellen](#)

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 bedeutet, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das `Action` Element 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, für die nur eine Genehmigung erforderlich ist und für die es keinen entsprechenden Vorgang gibt. API 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 Zugriff auf was 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. Sie können einem IAM Benutzer beispielsweise nur dann Zugriff auf eine Ressource gewähren, wenn sie mit seinem IAM Benutzernamen gekennzeichnet ist. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter [IAMRichtlinienelemente: Variablen und Tags](#).

AWS unterstützt globale Bedingungsschlüssel und dienstspezifische Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter [Kontext-Schlüssel für AWS globale Bedingungen](#) im IAMBenutzerhandbuch.

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.

ACLsin MediaTailor

UnterstütztACLs: 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

Unterstützungen ABAC (Tags in Richtlinien): Teilweise

Die attributbasierte Zugriffskontrolle (ABAC) ist eine Autorisierungsstrategie, bei der Berechtigungen auf der Grundlage von Attributen definiert werden. In werden AWS 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, die Operationen zulassen, wenn das Tag des Prinzipals mit dem Tag auf der Ressource übereinstimmt, auf die er zugreifen möchte.

ABAC ist hilfreich in Umgebungen, die schnell wachsen, und hilft in Situationen, in denen die Richtlinienverwaltung umständlich 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 finden Sie ABAC unter [Was ist? ABAC](#) im IAM Benutzerhandbuch. Ein Tutorial mit Schritten zur Einrichtung finden Sie im ABAC Benutzerhandbuch unter [Verwenden der attributbasierten Zugriffskontrolle \(ABAC\)](#). IAM

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 darüber, AWS -Services wie Sie mit temporären Anmeldeinformationen [arbeiten können AWS -Services](#), finden Sie IAM im IAM Benutzerhandbuch.

Sie verwenden temporäre Anmeldeinformationen, wenn Sie sich mit einer anderen AWS Management Console Methode als einem Benutzernamen und einem Kennwort 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 Rollenwechsel finden Sie unter [Wechseln zu einer Rolle \(Konsole\)](#) im IAMBenutzerhandbuch.

Mit dem AWS CLI oder können Sie manuell temporäre Anmeldeinformationen erstellen AWS API. 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 unter IAM](#).

Serviceübergreifende Prinzipalberechtigungen für MediaTailor

Unterstützt Forward-Access-Sitzungen (FAS): Ja

Wenn Sie einen IAM Benutzer oder eine 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. FASverwendet die Berechtigungen des Prinzipals, der einen aufruft AWS -Service, kombiniert mit der Anforderung, Anfragen AWS -Service an nachgelagerte Dienste zu stellen. FASANfragen 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 beim Stellen von 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 dienstbezogene 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

Standardmäßig sind Benutzer und Rollen nicht berechtigt, MediaTailor Ressourcen zu erstellen oder zu ändern. Sie können auch keine Aufgaben mithilfe von AWS Management Console, AWS Command Line Interface (AWS CLI) oder ausführen AWS API. Um Benutzern die Berechtigung zu erteilen, Aktionen mit den Ressourcen durchzuführen, die sie benötigen, kann ein IAM Administrator IAM Richtlinien erstellen. Der Administrator kann dann die IAM Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen übernehmen.

Informationen zum Erstellen einer IAM identitätsbasierten Richtlinie anhand dieser JSON Beispieldokumente finden Sie unter [IAMRichtlinien erstellen](#) im IAMBenutzerhandbuch.

Weitere Informationen zu Aktionen und Ressourcentypen, die von definiert wurden MediaTailor, einschließlich des Formats ARNs für die einzelnen 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 AWS im IAMBenutzerhandbuch unter [AWS Verwaltete Richtlinien oder Verwaltete Richtlinien für Jobfunktionen](#).

- Berechtigungen mit den geringsten Rechten anwenden — Wenn Sie Berechtigungen mit IAM Richtlinien festlegen, gewähren Sie nur die Berechtigungen, die für die Ausführung 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 IAM zum Anwenden von Berechtigungen finden Sie [IAMim Benutzerhandbuch unter Richtlinien und Berechtigungen](#). IAM
- Verwenden Sie Bedingungen in IAM Richtlinien, um den Zugriff weiter einzuschränken — Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen einzuschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben, um anzugeben, dass alle Anfragen über gesendet werden müssenSSL. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese über einen bestimmten Zweck verwendet werden AWS -Service, z. AWS CloudFormation B. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter [IAMJSONRichtlinienelemente: Bedingung](#).
- Verwenden Sie IAM Access Analyzer, um Ihre IAM Richtlinien zu validieren, um sichere und funktionale Berechtigungen zu gewährleisten. IAM Access Analyzer validiert neue und bestehende Richtlinien, sodass die Richtlinien der IAM Richtlinienensprache (JSON) und den IAM bewährten Methoden entsprechen. IAMAccess Analyzer bietet mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen, um Sie bei der Erstellung sicherer und funktionaler Richtlinien zu unterstützen. Weitere Informationen finden Sie unter [IAMAccess Analyzer-Richtlinienvvalidierung](#) im IAMBenutzerhandbuch.
- Multi-Faktor-Authentifizierung erforderlich (MFA) — Wenn Sie ein Szenario haben, in dem IAM Benutzer oder ein Root-Benutzer erforderlich sind AWS-Konto, aktivieren Sie die Option MFA für zusätzliche Sicherheit. Um festzulegen, MFA wann API Operationen aufgerufen werden, fügen Sie MFA Bedingungen zu Ihren Richtlinien hinzu. Weitere Informationen finden Sie unter [Konfiguration des MFA -geschützten API Zugriffs](#) im IAMBenutzerhandbuch.

Weitere Informationen zu bewährten Methoden finden Sie unter [Bewährte Sicherheitsmethoden IAM im IAM](#) Benutzerhandbuch. IAM

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 Anrufe an AWS CLI oder am tätigen, keine Mindestberechtigungen für die Konsole gewähren AWS API. Erlauben Sie stattdessen nur den Zugriff auf die Aktionen, die dem API Vorgang entsprechen, den sie 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 im [Benutzerhandbuch unter Hinzufügen von Berechtigungen für einen IAM](#) Benutzer.

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

Dieses Beispiel zeigt, wie Sie eine Richtlinie erstellen könnten, die es IAM Benutzern ermöglicht, die internen und verwalteten Richtlinien einzusehen, die mit ihrer Benutzeridentität verknüpft sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe von oder. AWS CLI AWS 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.

[Kanäle erstellen](#)

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 tritt auf, wo *region* ist der AWS-Region, *accountID* ist deine AWS-Konto ID, und *channelName* ist der Name der Kanalressource.

```

{
  "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 tritt auf, wo *region* ist der AWS-Region, *accountID* ist deine AWS-Konto ID, und *channelName* ist der Name der Kanalressource.

```

{
  "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 vom [IAMKunden verwaltete Richtlinien zu erstellen](#), die Ihrem Team nur die Berechtigungen gewähren, die es benötigt. 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 IAMBenutzerhandbuch 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 eine Beschreibung der Richtlinien für Jobfunktionen finden Sie im IAMBenutzerhandbuch unter [AWS Verwaltete Richtlinien für Jobfunktionen](#).

AWS verwaltete Richtlinie: `AWSElementalMediaTailorFullAccess`

Sie können die `AWSElementalMediaTailorFullAccess` Richtlinie an Ihre IAM Identitäten anhängen. Dies ist nützlich für Benutzer, die Wiedergabekonfigurationen und Ressourcen zur Kanalausammenstellung 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.

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": "mediatailor:*",
    "Resource": "*"
  }
}
```

AWS verwaltete Richtlinie: `AWSElementalMediaTailorReadOnly`

Sie können die `AWSElementalMediaTailorReadOnly` Richtlinie an Ihre IAM Identitäten anhängen. Es ist nützlich für Benutzer, die Wiedergabekonfigurationen und Ressourcen zur Kanalausammenstellung wie Programme und Kanäle einsehen müssen. Diese Richtlinie gewährt Berechtigungen, die nur Lesezugriff auf ermöglichen. AWS Elemental MediaTailor Diese Benutzer können keine Ressourcen erstellen, aktualisieren oder löschen MediaTailor .

```
{
  "Version": "2012-10-17",
  "Statement": {
```

```

"Effect": "Allow",
"Action": [
  "mediatailor:List*",
  "mediatailor:Describe*",
  "mediatailor:Get*"
],
"Resource": "*"
}
}

```

MediaTailor Aktualisierungen AWS verwalteter Richtlinien

Hier finden Sie Informationen zu Aktualisierungen der AWS verwalteten Richtlinien, die MediaTailor seit Beginn der Nachverfolgung dieser Änderungen durch diesen Dienst vorgenommen wurden. Abonnieren Sie den RSS Feed auf der, um automatische Benachrichtigungen über Änderungen an dieser Seite zu erhalten MediaTailor [Dokumenthistorie für AWS Elemental MediaTailor](#).

| Änderung | Beschreibung | Datum |
|---|---|-------------------|
| MediaTailor neue verwaltete Richtlinien hinzugefügt | MediaTailor hat die folgenden verwalteten Richtlinien hinzugefügt: <ul style="list-style-type: none"> • 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) [dienstbezogene Rollen](#). Eine dienstbezogene Rolle ist ein einzigartiger Rollentyp, mit dem direkt verknüpft ist. IAM 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. Zu den definierten Berechtigungen gehören die Vertrauensrichtlinie und die Berechtigungsrichtlinie, und diese Berechtigungsrichtlinie kann keiner anderen IAM Entität zugeordnet werden.

Sie können eine serviceverknüpfte Rolle erst löschen, nachdem ihre verwandten Ressourcen gelöscht wurden. Dadurch werden Ihre MediaTailor Ressourcen geschützt, da Sie die Zugriffsberechtigung für die Ressourcen nicht versehentlich entfernen können.

Informationen zu anderen Diensten, die dienstverknüpfte Rollen unterstützen, finden Sie unter [AWS Dienste, die mit Diensten funktionieren, IAM](#) und suchen Sie in der Spalte Dienstverknüpfte Rolle nach den Diensten, für die Ja steht. Wählen Sie über einen Link Ja aus, um die Dokumentation zu einer serviceverknüpften Rolle für diesen Service anzuzeigen.

Berechtigungen für dienstverknüpfte Rollen 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 `AWSServiceRoleForMediaTailor` 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 IAM Entität (z. B. ein Benutzer, eine Gruppe oder eine Rolle) eine dienstbezogene Rolle erstellen, bearbeiten oder löschen kann. Weitere Informationen finden Sie unter [Berechtigungen für dienstverknüpfte Rollen](#) im IAMBenutzerhandbuch.

Erstellen einer dienstverknüpften Rolle für MediaTailor

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. Wenn Sie die Sitzungsprotokollierung aktivieren, MediaTailor erstellt der AWS Management Console, AWS Command Line Interface (AWS CLI) oder der AWS API, die serviceverknüpfte Rolle für Sie.

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, wenn Sie den MediaTailor Dienst vor dem 15. September 2021 genutzt haben, als er begann, dienstbezogene Rollen zu unterstützen, dann MediaTailor haben Sie die AWSServiceRoleForMediaTailor Rolle in Ihrem Konto erstellt. Weitere Informationen findest du unter [Eine neue Rolle wurde in „Mein IAM Konto“](#) angezeigt.

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 die IAM Konsole auch verwenden, um eine dienstverknüpfte Rolle mit dem MediaTailorAnwendungsfall zu erstellen. Erstellen Sie im AWS CLI oder im AWS API eine dienstverknüpfte Rolle mit dem `mediatailor.amazonaws.com` Dienstnamen. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter [Erstellen einer dienstbezogenen Rolle](#). Wenn Sie diese serviceverknüpfte Rolle löschen, können Sie mit demselben Verfahren die Rolle erneut erstellen.

Bearbeiten einer serviceverknüpften 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 die Beschreibung der Rolle jedoch mithilfe IAM von bearbeiten. Weitere Informationen finden Sie unter [Bearbeiten einer dienstbezogenen Rolle](#) im IAMBenutzerhandbuch.

Löschen einer serviceverknüpften Rolle für MediaTailor

Wenn Sie ein Feature oder einen Dienst, die bzw. der eine serviceverknüpfte 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 serviceverknüpfte 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 `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 für „Prozent aktiviert“ auf 0. Dadurch wird die gesamte Sitzungsprotokollierung der entsprechenden Wiedergabekonfiguration deaktiviert. Weitere Informationen finden Sie unter [Deaktivierung einer Protokollkonfiguration](#).

Um die mit dem Dienst verknüpfte Rolle manuell zu löschen, verwenden Sie IAM

Verwenden Sie die IAM Konsole, die AWS Command Line Interface (AWS CLI) oder die, AWS API um die `AWSServiceRoleForMediaTailor` dienstverknüpfte Rolle zu löschen. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter [Löschen einer dienstverknüpften Rolle](#).

Unterstützte Regionen für MediaTailor dienstverknüpfte Rollen

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](#).

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 auftreten könnenIAM.

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 `mateojackson` IAM Benutzer versucht, die Konsole zu verwenden, um Details zu einer fiktiven `my-example-widget` Ressource anzuzeigen, aber nicht über die fiktiven `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 eine Fehlermeldung erhalten, dass Sie nicht berechtigt sind, die `iam:PassRole` Aktion auszuführen, müssen Ihre Richtlinien aktualisiert werden, damit Sie eine Rolle an MediaTailor diese Person übergeben 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 darüber, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie [im IAM Benutzerhandbuch unter Gewähren des Zugriffs auf einen anderen IAMBenutzer AWS-Konto , dessen Eigentümer Sie sind.](#)
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie [AWS-Konten im IAMBenutzerhandbuch unter Gewähren des Zugriffs für Dritte.](#)
- Informationen dazu, wie Sie Zugriff über einen Identitätsverbund [gewähren, finden Sie im Benutzerhandbuch unter Zugriff für extern authentifizierte Benutzer \(Identitätsverbund\).](#) IAM
- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontenübergreifenden Zugriff finden Sie [IAMim Benutzerhandbuch unter Kontoübergreifender Ressourcenzugriff.](#) IAM

Überprüfung der Einhaltung von Vorschriften 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:

- [Schnellstartanleitungen zu Sicherheit und Compliance](#) — In diesen Bereitstellungsleitfäden werden architektonische Überlegungen erörtert und Schritte für die Implementierung von Basisumgebungen beschrieben AWS , bei denen Sicherheit und Compliance im Mittelpunkt stehen.
- [Architecting for HIPAA Security and Compliance on Amazon Web Services](#) — In diesem Whitepaper wird beschrieben, wie Unternehmen Anwendungen erstellen HIPAA können, die AWS für sie in Frage kommen.

Note

Nicht alle sind berechtigt AWS -Services . HIPAA Weitere Informationen finden Sie in der [Referenz für HIPAA qualifizierte Dienste](#).

- [AWS Ressourcen zur AWS](#) von Vorschriften — Diese Sammlung von Arbeitsmapen und Leitfäden kann auf Ihre Branche und Ihren Standort zutreffen.
- [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 für 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)) zusammengefasst.

- [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](#)— Dies AWS -Service bietet 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. Eine Liste der unterstützten Services und Kontrollen finden Sie in der [Security-Hub-Steuerungsreferenz](#).
- [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 zu erfüllen PCIDSS, z. B. durch die Erfüllung der Anforderungen zur Erkennung von Eindringlingen, die in bestimmten Compliance-Frameworks vorgeschrieben sind.
- [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, die Sie bei Ihren Anforderungen an Datenstabilität und Datensicherung unterstützen.

Sicherheit der Infrastruktur in AWS Elemental 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:

- Sicherheit auf Transportschicht (TLS). Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Cipher-Suites mit perfekter Vorwärtsgeheimhaltung (PFS) wie (Ephemeral Diffie-Hellman) oder DHE (Elliptic Curve Ephemeral Diffie-Hellman). ECDHE Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi.

Darüber hinaus müssen Anfragen mithilfe einer Zugriffsschlüssel-ID und eines geheimen Zugriffsschlüssels, der einem Prinzipal zugeordnet ist, signiert werden. IAM 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 (AmazonVPC) -Endpunkten oder bestimmten VPCs zu kontrollieren. Dadurch wird der Netzwerkzugriff auf eine bestimmte MediaTailor Ressource effektiv nur von den spezifischen Ressourcen 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 wirksamste Schutz vor dem Problem mit dem verwirrten Stellvertreter besteht darin, den Kontextschlüssel „`aws:SourceArnGlobal Condition`“ mit ARN der gesamten Ressource zu verwenden. Wenn Sie die gesamte ARN Ressource nicht kennen oder wenn Sie mehrere Ressourcen angeben, verwenden Sie den `aws:SourceArn` globalen Kontextbedingungsschlüssel mit Platzhaltern (*) für die unbekanntenen Teile von. ARN Beispiel, `arn:aws:service:region:account_ID:*`.

Das folgende Beispiel zeigt, wie Sie die Kontextschlüssel `aws:SourceArn` und die `aws:SourceAccount` globale Bedingung verwenden können, um das Problem mit dem verwirrten Stellvertreter 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"
      }
    }
  }
}
```


}

Einloggen und Überwachen AWS Elemental 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 [Überwachung und Tagging](#).

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 SNS Amazon-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 [Protokollierung mit AWS CloudTrail](#).

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](#).

Überwachung und Tagging

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 Nutzung 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 Anzeigenentscheidungsserver (ADS) überwachen, speichern und darauf zugreifen. AWS Elemental MediaTailor sendet Protokolle für Anzeigenanfragen, Weiterleitungen und Antworten sowie für Berichtsfragen und -antworten aus. Fehler von den Servern ADS und den 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 dazu, wie Sie über Amazon CloudWatch Logs Insights auf sie zugreifen können, finden Sie unter [Protokolle anzeigen und abfragen ADS](#).

Themen

- [Berechtigungen für Amazon einrichten CloudWatch](#)
- [Überwachung mithilfe von CloudWatch Protokollen](#)
- [Überwachung AWS Elemental MediaTailor mit CloudWatch Amazon-Metriken](#)
- [Protokollierung mit AWS CloudTrail](#)
- [Überwachung der Ressourcen zur Kanalmontage mit MediaTailor Warnungen](#)

Berechtigungen für Amazon einrichten CloudWatch

Verwenden Sie AWS Identity and Access Management (IAM), um eine Rolle zu erstellen, die AWS Elemental MediaTailor Zugriff auf Amazon gewährt CloudWatch. 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 die IAM-Konsole unter <https://console.aws.amazon.com/iam/>.
2. Wählen Sie im Navigationsbereich der IAM Konsole Rollen und anschließend Rolle erstellen aus.
3. Wählen Sie den Rollentyp Anderes AWS Konto aus.
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).

Überwachung mithilfe von CloudWatch Protokollen

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.

Themen

- [Steuerung des Volumens der Sitzungsprotokolle zur Anzeigenschaltung](#)
- [Debug-Protokolle werden generiert](#)
- [Als Run-Protokoll für Channel Assembly](#)
- [Protokolle anzeigen und abfragen ADS](#)

Steuerung des Volumens der Sitzungsprotokolle zur Anzeigenschaltung

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 festlegen 60, 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 mit der MediaTailor Konsole oder den Wert für die Aktivierung in Prozent auf 0. API 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.

Debug-Protokolle werden generiert

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 die Debug-Protokolldaten in Logs MediaTailor geschrieben. CloudWatch Die Debug-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 zur MediaTailor Interaktion mit dem Ursprungsserver. Zum Beispiel die Antwort auf das Ursprungs-Manifest, der Manifesttyp und der UrsprungURL.
- 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.

Voraussetzungen

Um den Protokollmodus auf Debug zu setzen, 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

Für serverseitige Berichte fügen Sie den `?aws.LogMode=DEBUG` Abfrageparameter und den Wert in die Wiedergabeanforderung Ihres GET HTTP Players an den HLS DASH MediaTailor OR-Endpunkt ein. 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> | <p>Der Typ des Debug-Protokollereignisses.</p> <p>Werte:</p> <ul style="list-style-type: none"> <code>ORIGIN_INTERACTION</code> — Details zur MediaTailor Interaktion mit dem Ursprungsserver. Zum Beispiel die Antwort auf das Ursprungsmanifest, der Manifesttyp und der UrsprungURL. <code>GENERATED_MANIFEST</code> — Details zur Antwort auf die Wiedergabesitzung von MediaTailor. Zum Beispiel das Manifest, das MediaTailor generiert wird. <code>SESSION_INITIALIZED</code> — Details zur Sitzungsinitialisierung, z. B. die Sitzungs-ID. |
| <code>originRequestUrl</code> | Der 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. |

| Feld | Beschreibung |
|--------------|---|
| requestId | Die ID einer bestimmten HTTP Anfrage an MediaTailor. |
| responseBody | Das Manifest im Antworttext von MediaTailor. Dies ist entweder das rohe Ursprungsmanifest oder das von generierte Manifest MediaTailor. |
| sessionId | Die ID der Wiedergabe-Sitzung. |
| sessionType | Die Art der Wiedergabesitzung. 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-Logs 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"
```

Als Run-Protokoll für 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 vom Programm alternateMedia erstellten Regeln nicht unterstützt. Dies bedeutet, dass das As Run-Protokoll für derzeit nicht generiert wirdalternateMedia.

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 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. Wähle 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": []  
}
```

Protokolle anzeigen und abfragen ADS

Mit Amazon CloudWatch Logs Insights können Sie die Protokolle des AWS Elemental MediaTailor Ad Decision Servers (ADS) anzeigen und abfragen. MediaTailor sendet Ereignisprotokolle an, CloudWatch um normale Verarbeitungs- und Fehlerbedingungen zu gewährleisten. Die Protokolle folgen einem JSON Schema. Mit CloudWatch Logs Insights können Sie Protokolle nach Zeiträumen 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. Anweisungen finden Sie unter [Berechtigungen für Amazon einrichten CloudWatch](#).

Um ADS Protokolle mit der 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 **AdDec** ein, und wählen Sie dann aus der Dropdown-Liste 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 finden MediaTailor ADS Sie unter [Abfragen der Protokolle ADS](#).
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 Aktionen und dann Abfrageergebnisse herunterladen (CSV).

Note

Die Konsole begrenzt die Anzahl der Datensätze, die sie in Abfrageergebnissen zurückgibt und die sie exportiert. Verwenden Sie daher für Massendaten dasAPI, das AWS Command Line Interface (AWS CLI) oder einSDK.

Themen

- [ADSBeschreibung des Protokolls](#)
- [Abfragen der Protokolle ADS](#)
- [ADSProtokollschema JSON](#)

ADSBeschreibung des Protokolls

In diesem Abschnitt werden die Struktur und der Inhalt der ADS Protokollbeschreibung beschrieben. Verwenden Sie die Liste unter, um in einem JSON Editor auf eigene Faust zu forschen [the section called “ADSJSONProtokollschema”](#).

Jedes Ereignis im ADS Protokoll enthält die Standardfelder, die von CloudWatch Logs generiert werden. Weitere Informationen finden Sie unter [Analysieren von Protokoll Daten mit CloudWatch Logs Insights](#).

ADSEigenschaften von Protokollen

In diesem Abschnitt werden die Eigenschaften der ADS Protokolle beschrieben.

ADSEigenschaften der Protokolle

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|---------------|---------------------------------------|--------------|--|
| adsRequestUrl | Zeichenfolge | false | Die URL vollständige ADS Anfrage von MediaTailor. |
| avail | Objekt des Typs avail | false | Informationen zu einem Angebot, das voller MediaTailor Anzeigen 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 |

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|-------------------------------|--------------|--------------|--|
| | | | 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 Hash-Version der AWS Konto-ID, mit der Sie mehrere Protokolleinträge korrelieren können. |
| <code>eventDescription</code> | Zeichenfolge | true | Eine kurze Beschreibung des Ereignisses, das diese Protokollnachricht ausgelöst hat, bereitgestellt vom MediaTailor Dienst. Standardmäßig ist dieses Feld leer. Beispiel: Got VAST response. |
| <code>eventTimestamp</code> | Zeichenfolge | true | Das Datum und die Uhrzeit des Ereignisses. |
| <code>eventType</code> | Zeichenfolge | true | Der Code für das Ereignis, das diese Protokollnachricht ausgelöst hat. Beispiel: VAST_RESPONSE . |

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|-----------------------------|--|--------------|---|
| <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>requestHeaders</code> | Array vom Typ requestheaders | false | Die Header, die MediaTailor in der ADS Anfrage enthalten waren. In der Regel enthalten die Protokolle diese, wenn eine Anfrage an die ADS fehlschlägt, um bei der Problembearbeitung zu helfen. |
| <code>requestId</code> | Zeichenfolge | true | Die MediaTailor Anforderungs-ID, mit der Sie mehrere Protokolleinträge für dieselbe Anfrage korrelieren können. |

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|--------------------|--|--------------|--|
| sessionId | 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 . |
| sessionType | Zeichenfolge (zulässige Werte: [DASH,HLS]) | true | Der Stream-Typ des Spielers. |
| vastAd | Objekt des Typs vastAd | false | Informationen zu einer einzelnen Anzeige, die anhand der VAST Antwort analysiert wurden. |
| vastResponse | Objekt des Typs vastResponse | false | Informationen über die VAST Antwort, die von der MediaTailor ADS erhalten 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. |

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|---------------------------|------|--------------|---|
| vodVastResponseTimeOffset | Zahl | false | Der VMAP spezifische Zeitversatz für die VOD Anzeigeneinfügung. |

adContent

In diesem Abschnitt werden die Eigenschaften der ADS Protokolle beschrieben adContent.

ADS adContent Eigenschaften der Protokolle

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|---------------|---|--------------|--|
| adPlaylistUri | Objekt des Typs adPlaylistUri | false | Die Zuordnung vom Ursprungsmanifest für eine Variante zum Anzeigenmanifest für die Variante. Für DASH enthält dies einen einzigen Eintrag, da alle Varianten in einem einzigen DASH Manifest dargestellt werden. |

adPlaylistUri

In diesem Abschnitt werden die Eigenschaften der ADS Protokolle beschrieben adPlaylistUri.

ADS adPlaylistUris Eigenschaften der Protokolle

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|--------------|--------------|--------------|--|
| <any string> | Zeichenfolge | false | Das URL Anzeigenmanifest für die spezifische Variante. |

avail

In diesem Abschnitt werden die Eigenschaften der verfügbaren ADS Protokolle beschrieben.

ADSProtokolle verfügen über Eigenschaften

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|----------------|--|--------------|--|
| availId | Zeichenfolge | true | Die eindeutige Kennung für diese avail-Protokoll. Denn HLS dies ist die Mediensequenznummer, mit der die Verfügbarkeit beginnt. Für DASH ist dies die Perioden-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 |

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|----------------------------|---|--------------|--|
| | | | 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). |
| 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 beschriebencreativeAd.

ADS creativeAd Eigenschaften der Protokolle

| 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 den Creative in der VAST Antwort, falls verfügbar. Andernfalls ist es das Zwischengeschoss 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 BeaconsURLs. |
| transcodedAdDuration | Zahl | true | Die Dauer der Anzeige, berechnet aus der transcodierten Komponente. |

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|---------------------------|--------------|--------------|---|
| <code>uri</code> | Zeichenfolge | true | Die Mezzanine-Version URL der Anzeige, bei der es sich um die Eingabe für den Transcoder handelt. |
| <code>vastDuration</code> | Zahl | true | Die Dauer der Anzeige, wie sie anhand der Antwort analysiert wurde. VAST |

requestheaders

In diesem Abschnitt werden die Eigenschaften der RequestHeader der ADS Logs beschrieben.

ADSEigenschaften von Protokollen, RequestHeadern

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|--------------------|--------------|--------------|---------------------------------------|
| <code>name</code> | Zeichenfolge | true | Der Name des hinzuzufügenden Headers. |
| <code>value</code> | Zeichenfolge | true | Der Wert des Headers |

skippedAd

In diesem Abschnitt werden die Eigenschaften der ADS Protokolle beschrieben skippedAd.

ADS skippedAd Eigenschaften der Protokolle

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|-----------------------------|--------------|--------------|--------------------------|
| <code>adMezzanineUrl</code> | Zeichenfolge | true | Das Zwischengeschoss der |

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|----------------------------------|--------------|--------------|---|
| | | | übersprungenen URL Anzeige. |
| <code>creativeUniqueId</code> | 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 ist es das Zwischengeschoss URL der Anzeige. |
| <code>skippedReason</code> | Zeichenfolge | true | Der Code, der angibt, warum die Anzeige nicht eingefügt wurde. Beispiel: <code>TRANSCODE_IN_PROGRESS</code> . |
| <code>transcodeAdDuration</code> | Zahl | false | Die Dauer der Anzeige, berechnet aus der transcodierten Komponente. |
| <code>vastDuration</code> | Zahl | true | Die Dauer der Anzeige, wie sie anhand der Antwort analysiert wurde. VAST |

slateAd

In diesem Abschnitt werden die Eigenschaften der ADS Protokolle beschrieben `slateAd`.

ADS `slateAd` Eigenschaften der Protokolle

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|----------------------------------|---|--------------|---|
| <code>adContent</code> | Objekt des Typs adContent | true | Informationen über den Inhalt der eingefügten Anzeige. |
| <code>creativeUniqueId</code> | Zeichenfolge | true | Die eindeutige Kennung für die Anzeige, die als Schlüssel für die Transcodierung verwendet wird. Dies ist das ID-Feld für den Creative in der VAST Antwort, falls verfügbar. Andernfalls ist es das Zwischengeschoss URL der Anzeige. |
| <code>transcodeAdDuration</code> | Zahl | true | Die Dauer der Anzeige, berechnet aus der transcodierten Komponente. |
| <code>uri</code> | Zeichenfolge | true | Die Mezzanin-Version URL der Anzeige, bei der es sich um die Eingabe für den Transcoder handelt. |

trackingEvents

In diesem Abschnitt werden die Eigenschaften der ADS Protokolle beschrieben trackingEvents.

ADS trackingEvents Eigenschaften der Protokolle

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|--------------|----------------------------|--------------|--|
| <any string> | Array vom Typ Zeichenfolge | false | Die Liste der Beacons URLs für das angegebene Tracking-Ereignis (Impression, abgeschlossen usw.) |

vastAd

In diesem Abschnitt werden die Eigenschaften der ADS Protokolle beschrieben vastAd.

ADS vastAd Eigenschaften der Protokolle

| 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 |

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|-----------------------------|--|--------------|---|
| | | | Tags in der VAST Antwort. |
| <code>duration</code> | Zahl | true | Die ungefähre Dauer der Anzeige, basierend auf dem <code>duration</code> Tag im <code>linear</code> Element der VAST Antwort. |
| <code>trackingEvents</code> | 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 BeaconsURLs. |
| <code>vastAdId</code> | Zeichenfolge | true | Der Wert des ID-Attributs des Ad Tags in der Antwort VAST |
| <code>vastAdTagUri</code> | Zeichenfolge | false | Die VMAP -spezifische Weiterleitung URI für eine Anzeige. |
| <code>vastMediaFiles</code> | 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.

ADS vastMediaFile Eigenschaften der Protokolle

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|--------------|--------------|--------------|--|
| apiFramework | Zeichenfolge | true | Das API Framework, das für die Verwaltung der Mediendatei benötigt wird. Beispiel: VPAID. |
| bitrate | Zahl | true | Die Bitrate der Mediendatei. |
| delivery | Zeichenfolge | true | Das Protokoll, das für die Mediendatei verwendet wird, ist 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, der dem Typattribut des MediaFile Tags entnommen wurde. |
| uri | Zeichenfolge | true | Die URL Mezzanin-Version der Anzeige, die als Eingabe für den Transcoder dient. |

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|--------------------|------|--------------|----------------------------------|
| <code>width</code> | Zahl | true | Die Pixelbreite der Mediendatei. |

vastResponse

In diesem Abschnitt werden die Eigenschaften der ADS Protokolle beschrieben vastResponse.

ADS vastResponse Eigenschaften der Protokolle

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|----------------------|--------------------------------------|--------------|--|
| <code>errors</code> | Array vom Typ Zeichenfolge | true | Der Fehler URLs wurde anhand der Error Tags in der VAST Antwort analysiert. |
| <code>vastAds</code> | Array vom Typ vastAd | true | Die anhand der Antwort analysierten Anzeigen. VAST |
| <code>version</code> | Zeichenfolge | true | Die VAST Spezifikationsversion, die anhand des <code>version</code> Attributs des VAST Tags in der Antwort analysiert wurde. |

vodCreativeOffsets

In diesem Abschnitt werden die Eigenschaften der ADS Protokolle beschrieben vodCreativeOffsets.

ADS vodCreativeOffsets Eigenschaften der Protokolle

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|--------------|---|--------------|--|
| <any string> | 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.

ADS vodCreativeOffset Eigenschaften der Protokolle

| 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 den Creative in der VAST Antwort, falls verfügbar. Andernfalls ist es das Zwischengeschoss URL der Anzeige. |

| Eigenschaft | Typ | Erforderlich | Beschreibung |
|----------------------------------|--|--------------|---|
| <code>trackingEvents</code> | 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 BeaconsURLs. |
| <code>transcodeAdDuration</code> | Zahl | true | Die Dauer der Anzeige, berechnet aus der transcodierten Komponente. |
| <code>uri</code> | Zeichenfolge | true | Die Mezzanine-Version URL der Anzeige, bei der es sich um die Eingabe für den Transcoder handelt. |
| <code>vastDuration</code> | Zahl | true | Die Dauer der Anzeige, wie sie anhand der Antwort analysiert wurde. VAST |

Abfragen der Protokolle ADS

CloudWatch Logs Insights bietet eine Vielzahl von Optionen zum Abfragen Ihrer Protokolle. Ausführliche Informationen zur Abfragesyntax finden Sie unter [CloudWatch Logs Insights-Abfragesyntax](#). Dieser Abschnitt enthält Beispiele für häufig verwendete Abfragen, um Ihnen den

Einstieg in Ihre ADS Log-Abfragen zu erleichtern. 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 Anfragen an die ab. ADS Diese Abfrage zeigt eine Möglichkeit, den Inhalt des Anforderungsheaders für MediaTailor Protokolle 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 abURLs, 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)
```

ADSProtokollschema JSON

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",
        "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": {
            "$id": "#/properties/avail/fillRate",
            "type": "number",
            "description": "The rate at which the ads fill the avail duration, from 0.0
(for 0%) to 1.0 (for 100%).",
        },
        "creativeAds": {
            "$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",
                ]
            }
        }
    }

```



```

        "adContent",
        "trackingEvents",
        "vastDuration",
        "transcodedAdDuration"
    ],
    "additionalProperties": false,
    "properties": {
        "uri": { "$ref": "#/definitions/adMezzanineUri" },
        "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
        "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" }
      }
    }
  }
},

"vastResponse": {
  "$id": "#/properties/vastResponse",
  "type": "object",
  "title": "vastResponse",
  "description": "Information about the VAST response that MediaTailor received
from the ADS.",

```

```

    "required": [
      "version",
      "vastAds",
      "errors"
    ],
    "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": {
        "$id": "#/properties/vastResponse/errors",
        "type": "array",
        "description": "The error URLs parsed from the <code>Error</code> tags in the
VAST response.",
        "items": {
          "type": "string",
          "description": "A single error URL."
        }
      }
    }
  },
  "vastAd": {
    "$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."
},

"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."
      }
    }
  }
}
},

"oneOf": [
  { "$ref": "#/definitions/eventMakingAdsRequest" },
  { "$ref": "#/definitions/eventVastResponse" },
  { "$ref": "#/definitions/eventFilledAvail" },
  { "$ref": "#/definitions/eventErrorFiringBeaconFailed" },
  { "$ref": "#/definitions/eventWarningNoAdvertisements" },
  { "$ref": "#/definitions/eventUnknownHost" },
  { "$ref": "#/definitions/eventErrorAdsTimeout" },

```

```

    { "$ref": "#/definitions/eventPlannedAvail" },
    { "$ref": "#/definitions/eventEmptyVastResponse" },
    { "$ref": "#/definitions/eventErrorUnknown" },
    { "$ref": "#/definitions/eventVastRedirect" },
    { "$ref": "#/definitions/eventRedirectedVastResponse" },
    { "$ref": "#/definitions/eventErrorAdsResponseParse" },
    { "$ref": "#/definitions/eventErrorAdsInvalidResponse" },
    { "$ref": "#/definitions/eventErrorDisallowedHost"},
    { "$ref": "#/definitions/eventWarningDynamicVariableSubFailed"},
    { "$ref": "#/definitions/eventVodTimeBasedAvailPlanVastResponseForOffset" },
    { "$ref": "#/definitions/eventVodTimeBasedAvailPlanSuccess" }
  ],

```

```

"definitions": {
  "eventMakingAdsRequest": {
    "$id": "#/definitions/eventMakingAdsRequest",
    "required": [
      "eventType",
      "adsRequestUrl"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "MAKING_ADS_REQUEST"
      }
    }
  }
},

```

```

"eventVastResponse": {
  "$id": "#/definitions/eventVastResponse",
  "required": [
    "eventType"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "VAST_RESPONSE"
    }
  }
},

```

```

"eventFilledAvail": {
  "$id": "#/definitions/eventFilledAvail",

```

```
"required": [
  "eventType",
  "avail"
],
"properties": {
  "eventType": {
    "type": "string",
    "const": "FILLED_AVAIL"
  }
}
},

"eventErrorFiringBeaconFailed": {
  "$id": "#/definitions/eventErrorFiringBeaconFailed",
  "required": [
    "eventType",
    "error",
    "beaconInfo"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "ERROR_FIRING_BEACON_FAILED"
    }
  }
},

"eventWarningNoAdvertisements": {
  "$id": "#/definitions/eventWarningNoAdvertisements",
  "required": [
    "eventType"
  ],
  "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"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "PLANNED_AVAIL"
    }
  }
},

"eventEmptyVastResponse": {
  "$id": "#/definitions/eventEmptyVastResponse",
  "required": [
    "eventType"
  ],
  "properties": {
```



```
    "eventType": {
      "type": "string",
      "const": "EMPTY_VAST_RESPONSE"
    }
  }
},

"eventErrorUnknown": {
  "$id": "#/definitions/eventErrorUnknown",
  "required": [
    "eventType"
  ],
  "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"  
    ],  
    "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"  
      }  
    }  
  },  
  
  "eventErrorDisallowedHost": {  
    "$id": "#/definitions/eventErrorDisallowedHost",  
    "required": [  
      "eventType"  
    ],  
    "properties": {  
      "eventType": {  
        "type": "string",  
        "const": "ERROR_DISALLOWED_HOST"  
      }  
    }  
  },  
}
```

```

"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. "
  },

  "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."
  },
}
```

```

"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."
    }
  }
},

"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": {

```

```

    "$id": "#/properties/vastAd/adSystem",
    "type": "string",
    "description": "The value of the <code>AdSystem</code> tag in the VAST
response.",
    "examples": [
      "GDFP"
    ]
  },
  "adTitle": {
    "$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": {
        "$id": "#/properties/vastAd/vastMediaFiles/items/properties/
apiFramework",
        "type": "string",
        "description": "The API framework needed to manage the media file.
Example: <code>VPAID</code>."
    },
    "width": {
        "$id": "#/properties/vastAd/vastMediaFiles/items/properties/width",
        "type": "integer",
        "description": "The pixel width of the media file.",
        "examples": [
            1280
        ]
    },
    "height": {
        "$id": "#/properties/vastAd/vastMediaFiles/items/properties/height",
        "type": "integer",
        "description": "The pixel height of the media file.",
        "examples": [
            720
        ]
    },
    "bitrate": {
        "$id": "#/properties/vastAd/vastMediaFiles/items/properties/bitrate",
        "type": "integer",
        "description": "The bitrate of the media file.",
        "examples": [
            533
        ]
    }
}
},
"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"  
      ]  
    }  
  }  
}  
}
```

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 Verlaufsdaten 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 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 |
|----------------------------|---|
| <code>4xxErrorCount</code> | Die Anzahl der 4xx Fehler. |
| <code>5xxErrorCount</code> | Die Anzahl der 5xx Fehler. |
| <code>RequestCount</code> | 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. |
| <code>TotalTime</code> | Die Zeit, die der Anwendungsserver für die Bearbeitung der Anfrage benötigte, einschließlich der Zeit, die für den Empfang von Bytes vom Client und das Schreiben von Bytes in den Client und das Netzwerk benötigt wurde. |

Metriken zum serverseitigen Einfügen von Werbung () SSAI

In der folgenden Tabelle sind Metriken zur serverseitigen Anzeigeneinfügung aufgeführt.

| Metrik | Beschreibung |
|-----------------------------------|---|
| <code>AdDecisionServer.Ads</code> | Die Anzahl der Anzeigen, die innerhalb des von Ihnen angegebenen CloudWatch Zeitraums in den Antworten des Ad Decision Servers (ADS) enthalten waren. |

| Metrik | Beschreibung |
|--|---|
| <code>AdDecisionServer.Duration</code> | Die Gesamtdauer aller Anzeigen in Millisekunden, die ADS innerhalb des von Ihnen angegebenen CloudWatch Zeitraums 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 Statuscode-Antworten, die nicht HTTP 200 sind, der leeren Antworten und der Antworten mit Timeout, die ADS innerhalb des von Ihnen CloudWatch angegebenen Zeitraums MediaTailor eingegangen sind. |
| <code>AdDecisionServer.FillRate</code> | <p>Der einfache Durchschnitt der Raten, mit denen die Antworten der ADS ausgefüllten einzelnen Anzeige für den von Ihnen angegebenen Zeitraum verfügbar sind.</p> <p>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.</p> |
| <code>AdDecisionServer.Latency</code> | Die Antwortzeit in Millisekunden für Anfragen von an. MediaTailor ADS |
| <code>AdDecisionServer.Timeouts</code> | Die Anzahl der Anfragen, bei denen das Timeout ADS in dem von Ihnen angegebenen CloudWatch Zeitraum überschritten wurde. |

| Metrik | Beschreibung |
|----------------------|---|
| AdNotReady | <p>Die Häufigkeit, mit der auf eine Anzeige ADS verwiesen wurde, die in dem von Ihnen angegebenen Zeitraum noch nicht vom internen Transcoderdienst transkodiert wurde.</p> <p>Ein hoher Wert für diese Metrik kann zu einem niedrigen Gesamtwert für <code>Avail.FillRate</code> betragen.</p> |
| AdsBilled | <p>Die Anzahl der Anzeigen, für die Kunden auf der Grundlage der Insertion in MediaTailor Rechnung gestellt werden.</p> |
| Avail.Duration | <p>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.</p> |
| Avail.FilledDuration | <p>Die geplante Anzahl von Millisekunden an Werbeanzeigen, die innerhalb des Zeitraums mit MediaTailor Werbeanzeigen gefüllt werden. CloudWatch</p> |

| Metrik | Beschreibung |
|------------------|--|
| Avail.FillRate | <p>Der geplante einfache Durchschnitt der Raten, zu denen einzelne Anzeigen innerhalb des Zeitraums geschaltet MediaTailor werden.</p> <p>CloudWatch</p> <p>Um den gewichteten Durchschnittswert zu erhalten, berechnen Sie die <code>Avail.FillledDuration</code> als Prozentsatz der <code>Avail.Duration</code>. Weitere Informationen über einfache und gewichtete Durchschnittswerte finden Sie unter Einfache und gewichtete Durchschnittswerte.</p> <p>Das <code>MaximumAvail.FillRate</code>, das erreicht MediaTailor werden kann, ist begrenzt durch die <code>AdDecisionServer.FillRate</code>. Wenn der Wert für <code>Avail.FillRate</code> niedrig ist, vergleichen Sie ihn mit dem Wert für <code>AdDecisionServer.FillRate</code>. Wenn der <code>AdDecisionServer.FillRate</code> Wert niedrig ist, erhalten Sie ADS möglicherweise nicht genügend Anzeigen für die verfügbare Dauer.</p> |
| Avail.Impression | Die Anzahl der Anzeigen mit Impressions-Tracking-Ereignissen, die MediaTailor beim serverseitigen Beaconing erkannt werden (nicht die Anzahl der Impressionen). |

| Metrik | Beschreibung |
|---|---|
| <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. |
| <code>Avail.ObservedFilledDuration</code> | Die beobachtete Anzahl von Millisekunden der Werbeanzeige, die innerhalb des Zeitraums mit Werbeanzeigen MediaTailor gefüllt wurde. CloudWatch |
| <code>Avail.ObservedFillRate</code> | Der beobachtete einfache Durchschnitt der Raten, MediaTailor mit denen gefüllte einzelne Anzeigen innerhalb des Zeitraums geschaltet wurden. CloudWatch |
| <code>Avail.ObservedSlateDuration</code> | Die beobachtete Gesamtzahl der Millisekunden an Schiefereinträgen, die innerhalb des Zeitraums eingefügt wurden. CloudWatch |
| <code>GetManifest.Errors</code> | Die Anzahl der Fehler, die MediaTailor beim Generieren von Manifesten in dem von Ihnen angegebenen CloudWatch Zeitraum aufgetreten sind. |
| <code>GetManifest.Latency</code> | Die MediaTailor Antwortzeit in Millisekunden für die Anforderung zur Generierung von Manifesten. |
| <code>Origin.Errors</code> | Die Anzahl der Statuscode-Antworten, die nicht HTTP 200 sind, und der Antworten mit Timeout, die in dem von Ihnen angegebenen CloudWatch Zeitraum vom Ursprungsserver MediaTailor empfangen wurden. |

| Metrik | Beschreibung |
|--|--|
| <code>Origin.Latency</code> | Die Antwortzeit für Anfragen MediaTailor an Ihren Inhaltsursprungsserver. |
| <code>Origin.ManifestFileSizeBytes</code> | Die Dateigröße des Original-Manifests in Byte für HLS sowohl als auch DASH. In der Regel wird diese Metrik in Verbindung mit verwendet <code>Origin.ManifestFileSizeTooLarge</code> . |
| <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 verwendet <code>Origin.ManifestFileSizeBytes</code> . |
| <code>Origin.Timeouts</code> | Die Anzahl der Anfragen an den Ursprungsserver, bei denen das Timeout in dem von Ihnen CloudWatch angegebenen Zeitraum überschritten wurde. |
| <code>Requests</code> | Die Anzahl gleichzeitiger Transaktionen pro Sekunde für alle Anforderungstypen. Die Anzahl der Transaktionen hängt hauptsächlich 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-Version eingefügt wurden, weil eine Anzeigendauer ADS zurückgegeben wurde, die länger als die angegebene Gültigkeitsdauer war. Ein hoher Wert für diese Kennzahl kann zu einer Diskrepanz zwischen der Metrik und beitragen. <code>Avail.Ads</code> <code>AdDecisionServer.Ads</code> |

| Metrik | Beschreibung |
|---|---|
| <code>SkippedReason.EarlyCueIn</code> | Die Anzahl der Anzeigen, die aufgrund eines vorzeitigen Fehlers übersprungen wurden. CUE-IN |
| <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 für <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. |

| Metrik | Beschreibung |
|--|--|
| <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 für <code>Avail.FillRate</code> , bis die Inhalte erfolgreich transkodiert werden können. |

Einfache und gewichtete Durchschnittswerte

Sie können den einfachen Durchschnitt und den gewichteten Durchschnitt für die Antworten ADS auf die beiden 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 $(AdDecisionServer.Duration * 100) / Avail.Duration$ und $(Avail.FilledDuration * 100) / Avail.Duration$ 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 $(AdDecisionServer.Duration * 100) / Avail.Duration$ angegebenen gewichteten Durchschnitt. Der mit `Avail.FillRate` angegebene einfache Durchschnitt entspricht dem mit $(Avail.FilledDuration * 100) / Avail.Duration$ 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 auf die Verfügbarkeit umfasst 45 Sekunden lang Werbeanzeigen (50% gefüllt).
 - MediaTailor füllt 45 Sekunden der verfügbaren Werbezeit aus (50% gefüllt).
- Die Dauer des zweiten Ad-Avail beträgt 120 Sekunden:
 - Die ADS Antwort auf den Avail umfasst 120 Sekunden Werbung (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 Durchschnitt der ADS Füllraten liegt bei 78,57%, was einem Prozentsatz der `AdDecisionServer.Duration` folgenden Werte entspricht `Avail.Duration`: $(165 * 100) / 210$. 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 Füllquoten liegt bei 64,29%, was einem Prozentsatz der `Avail.FilledDuration` folgenden 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%. ADSEs kann sein, dass mehr Anzeigenzeit zurückgegeben wird, als im Angebot verfügbar ist, MediaTailor kann aber nur die verfügbare Zeit 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. |

Protokollierung mit AWS CloudTrail

AWS Elemental MediaTailor ist in einen Dienst integriert AWS CloudTrail, der eine Aufzeichnung der Aktionen bereitstellt, die von einem Benutzer, einer Rolle oder einem AWS Dienst in ausgeführt wurden MediaTailor. CloudTrail erfasst alle API Aufrufe MediaTailor als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der MediaTailor Konsole und Code-Aufrufe der 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 SNS Amazon-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 protokolliert CloudTrail und sind in der [AWS Elemental MediaTailor APIReferenz](#) dokumentiert. Beispielsweise generieren Aufrufe der `ListPlaybackConfigurations` Operationen `PutPlaybackConfiguration` und Einträge in den CloudTrail Protokolldateien.

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 den Anmeldeinformationen AWS Identity and Access Management (IAM) 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 userIdentity Element](#).

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 stellt eine einzelne Anforderung aus einer beliebigen Quelle dar und enthält unter anderem Informationen über die angeforderte Aktion, das Datum und die Uhrzeit der Aktion sowie über die Anforderungsparameter. CloudTrail Protokolldateien sind kein geordneter Stack-Trace der öffentlichen API Aufrufe, sie erscheinen also nicht in einer bestimmten Reihenfolge.

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://exampleleads.com"
},
"responseElements": {
  "SessionInitializationEndpointPrefix": "https://
bdaaeb4bd9114c088964e4063f849065.mediatailor.us-east-1.amazonaws.com/v1/session/
AKIAIOSFODNN7EXAMPLE/examplename/",
  "DashConfiguration": {
    "ManifestEndpointPrefix": "https://
bdaaeb4bd9114c088964e4063f849065.mediatailor.us-east-1.amazonaws.com/v1/dash/
AKIAIOSFODNN7EXAMPLE/examplename/",
    "MpdLocation": "EMT_DEFAULT"
  },
  "AdDecisionServerUrl": "http://exampleleads.com",
  "CdnConfiguration": {},
  "PlaybackEndpointPrefix": "https://
bdaaeb4bd9114c088964e4063f849065.mediatailor.us-east-1.amazonaws.com",
  "HlsConfiguration": {
    "ManifestEndpointPrefix": "https://
bdaaeb4bd9114c088964e4063f849065.mediatailor.us-east-1.amazonaws.com/v1/master/
AKIAIOSFODNN7EXAMPLE/examplename/"
  },
  "VideoContentSourceUrl": "http://examplevideo.com",
  "Name": "examplename"
},
"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": "examplename"
  },
  "responseElements": null,
  "requestID": "0z1y2x3w-0123-4567-9876-6q7r8s9t0u1v",
  "eventID": "888ddd77-3322-eeew-uuii-abc123jkl1343",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
}
```

Überwachung der Ressourcen zur Kanalmontage mit MediaTailor Warnungen

MediaTailor erstellt Warnmeldungen für Probleme oder potenzielle Probleme, die mit Ihren Channel-Assembly-Ressourcen auftreten. Die Warnung beschreibt das Problem, wann das Problem aufgetreten ist und welche Ressourcen betroffen sind.

Sie können die Benachrichtigungen in der AWS Management Console, der AWS Command Line Interface (AWS CLI), AWS SDKs oder programmgesteuert mit dem MediaTailor [ListAlerts API](#).

⚠ 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 | Warnungscode | Warnmeldung | Hinweise |
|-----------------|---------------------------|---|--|
| VOD-Quelle | NOT_PROCESSED | MediaTailor hat die Paketkonfiguration nicht verarbeitet <i>Konfigurationspfad</i> . | |
| | UNREACHABLE | Wir können die URL nicht erreichen <i>url</i> . | |
| | UNAUTHORIZED | <i>url</i> hat die Anfrage nicht autorisiert. | |
| | TIMEOUT | Die Verbindung zu <i>url</i> Zeitlimit überschritten. | |
| | UNPARSABLE_MANIFEST | MediaTailor Beim Analysieren des Manifests von ist ein Problem aufgetreten <i>url</i> . | |
| | VARIANT_DURATION_MISMATCH | MediaTailor stieß beim Analysieren des Manifests auf Varianten mit nicht übereinstimmender Gesamtdauer <i>url</i> . Dies kann zu Verzögerungen bei der Wiedergabe führen. | Ihr Manifest hat je nach Varianten/Repräsentation unterschiedliche Dauern. Dies kann zu fehlenden oder falschen Bildunterschriften führen, und MediaTailor keine Anzeigen einfügen können. |

| Art der Warnung | Warnungscode | Warnmeldung | Hinweise |
|-----------------|---------------------------|--|--|
| | SEGMENT_DURATION_TOO_LONG | MediaTailorist beim Analysieren des Manifests von auf ein Segment mit einer Dauer von mehr als dreißig Sekunden gestoßen <code>url</code> . Dies kann zu Verzögerungen bei der Wiedergabe, zu fehlenden oder falschen Untertiteln und zur Unfähigkeit, Werbung einzufügen, führen. | Ihr Manifest enthält ein Segment, das länger als 30 Sekunden ist. |
| | TARGET_DURATION_MISMATCH | MediaTailorstieß auf ein Missverhältnis vonEXT-X-TARGETDURATION Werte in HLS-Manifesten beim Analysieren des Manifests von <code>url</code> . Dies kann zu Verzögerungen bei der Wiedergabe führen. | Die Zieldauer stimmt nicht für alle Manifeste in der Quelle überein. |
| Ort der Quelle | NOT_PROCESSED | MediaTailorhat die Ressource nicht verarbeitet <code>Name der Ressource</code> . | |
| Programm | VOD_SOURCE_ALERT | Die VOD-Quelle <code>vodSourceName</code> in diesem Programm gibt es die folgende Warnung: <code>vodSourceAlertKode :vodSourceAlertNachricht</code> | |

| Art der Warnung | Warnungsc ode | Warnmeldung | Hinweise |
|-----------------|-------------------------------|---|----------|
| | SOURCE_LO CATION_AL ERT | Der Quellort <i>sourceLoc ationName</i> In diesem Programm enthalten ist die folgende Warnung: <i>sourceLoc ationAler tKode :sourceLoc ationAlertNachricht</i> | |
| | CODEC_MIS MATCH | MediaTailor stieß auf einen nicht übereinst immenden Codec in <i>Name des Kanals</i> zeitplan. Das Missverhältnis ist das <i>sourceGro upName</i> zwischen <i>Programm ame 1</i> ist Manifest <i>Manifest- URL</i> und <i>Programmname 2</i> ist Manifest <i>Manifest- URL</i> . | |
| | RESOLUTIO N_MISMATC H | MediaTailor stieß auf eine nicht übereinstimmende Auflösung in <i>Name des Kanals</i> zeitplan. Das Missverhältnis ist das <i>sourceGro upName</i> zwischen <i>Programm ame 1</i> ist Manifest <i>Manifest- URL</i> und <i>Programmname 2</i> ist Manifest <i>Manifest- URL</i> . | |

| Art der Warnung | Warnungscode | Warnmeldung | Hinweise |
|-----------------|--------------------|--|----------|
| | BANDWIDTH_MISMATCH | MediaTailorstieß auf eine nicht übereinstimmende Bandbreite in <i>Name des Kanals</i> zeitplan. Das Missverhältnis ist das <i>sourceGroup</i> <i>upName</i> zwischen <i>Programname 1</i> ist Manifest <i>Manifest-URL</i> und <i>Programname 2</i> ist Manifest <i>Manifest-URL</i> . | |
| | FRAMERATE_MISMATCH | MediaTailorstieß auf eine nicht übereinstimmende Framerate in <i>Kanalname</i> zeitplan. Das Missverhältnis ist das <i>sourceGroup</i> <i>upName</i> zwischen <i>Programname 1</i> ist Manifest <i>Manifest-URL</i> und <i>Programname 2</i> ist Manifest <i>Manifest-URL</i> . | |

| Art der Warnung | Warnungsc ode | Warnmeldung | Hinweise |
|-----------------|---------------------------|--|----------|
| | TARGET_DURATION_MISMATCH | MediaTailorstieß auf nicht übereinstimmendeEXT-X-TARGETDURATION Werte in HLS manifestieren sich in <i>Kanalname</i> zeitplan. Das Missverhältnis ist das <i>sourceGroupName</i> zwischen <i>Programmname 1</i> ist Manifest <i>Manifest-URL</i> und <i>Programmname 2</i> ist Manifest <i>Manifest-URL</i> . | |
| | SEGMENT_DURATION_MISMATCH | MediaTailorstieß auf nicht übereinstimmende Werte für die Segmentdauer in allen Manifesten in <i>Kanalname</i> zeitplan. Das Missverhältnis ist das <i>sourceGroupName</i> zwischen <i>Programmname 1</i> ist Manifest <i>Manifest-URL</i> und <i>Programmname 2</i> ist Manifest <i>Manifest-URL</i> . | |

| Art der Warnung | Warnungsc ode | Warnmeldung | Hinweise |
|-----------------|---|---|----------|
| | NO_COMMON_SEGMENT_BOUNDARY_FOR_AD_SLATE | MediaTailorwar nicht in der Lage, Werbung im Offset einzufügen <i>nOffset Millis</i> für das Programm <i>Name des Programms</i> . Zu Beginn der Werbeanzeige gibt es keine gemeinsame Segmentgrenze. | |
| | NOT_PROCESSED | MediaTailorhat die Ressource nicht verarbeitet <i>Name der Ressource</i> . | |
| | TOO_MANY_ALERTS | MediaTailorhat zu viele Alerts gefunden und gibt keine weiteren Alerts für <i>Name des Programms</i> . Löschen Sie bestehende Benachrichtigungen, um weiterhin Benachrichtigungen zu erhalten für <i>Name des Programms</i> . | |
| Channel | PROGRAM_ALERT | Das Programm <i>Name des Programms</i> In diesem Kanal enthalten ist die folgende Warnung: <i>programAlertCode :programAlertMessage</i> | |

Benachrichtigungen anzeigen

Sie können Benachrichtigungen für alle anzeigen *MediaTailorRessource* zur Kanalmontage. Wenn Sie sich die Benachrichtigungen für Kanäle und Programme ansehen, *MediaTailor* umfasst alle

zugehörigen Ressourcen, die im Kanal oder Programm enthalten sind. Wenn Sie sich beispielsweise die Benachrichtigungen für ein bestimmtes Programm ansehen, werden Ihnen auch Warnungen für den Quellort und die VOD-Quellen angezeigt, die das Programm enthält.

Gehen Sie wie folgt vor, um Warnmeldungen anzuzeigen.

Console

Um Benachrichtigungen in der Konsole anzuzeigen

1. Öffne die MediaTailor-Konsole bei <https://console.aws.amazon.com/mediatailor/>.
2. Wählen Sie die Ressource aus, für die Sie Benachrichtigungen anzeigen möchten.
3. Wählen Sie die Warnmeldungen-Tabulator, um die Benachrichtigungen anzuzeigen.

AWS Command Line Interface (AWS CLI)

Um Alerts für eine Channel-Assembly-Ressource aufzulisten, benötigen Sie die Ressource [Amazon-Ressourcenname \(ARN\)](#). Sie können das verwendende `describe-resource-type` Befehl in der AWS Command Line Interface (AWS CLI), um den ARN der Ressource abzurufen. Führen Sie zum Beispiel den [Kanal beschreiben](#) Befehl, um den ARN eines bestimmten Kanals abzurufen:

```
aws mediatailor describe-channel --channel-name MyChannelName
```

Dann benutze den [AWS MediaTailor-Listenbenachrichtigungen](#) Befehl zum Auflisten der mit der Ressource verknüpften Warnungen:

```
aws mediatailor list-alerts --resource-arn arn:aws:mediatailor:region:aws-account-id:resource-type/resource-name
```

API

Um Alerts für eine Channel-Assembly-Ressource aufzulisten, benötigen Sie die Ressource [Amazon-Ressourcenname \(ARN\)](#). Sie können das verwendende `DescribeResource` Operation in der MediaTailor API zum Abrufen des ARN der Ressource. Verwenden Sie zum Beispiel den [DescribeChannel](#) Vorgang, um den ARN eines bestimmten Kanals abzurufen.

Dann benutze den [ListAlerts](#) API zum Auflisten der Warnungen für die Ressource.

Umgang mit Alarmen

Wenn eine Warnung auftritt, sehen Sie sich die Warnmeldungen in der AWS Management Console, oder benutze den AWS Command Line Interface (AWS CLI), AWS SDKs oder die MediaTailor Benachrichtigt die API, um die möglichen Ursachen des Problems zu ermitteln.

Nachdem Sie das Problem gelöst haben, löscht MediaTailor den Alarm.

Markieren von AWS Elemental MediaTailor-Ressourcen

Ein Tag ist ein Metadaten-Etikett, das von Ihnen oder von AWS einer AWS-Ressource zugewiesen wird. 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:

- **Identify and organize your AWS resources.** Viele AWS-Services unterstützen das Markieren mit Tags (kurz: Tagging). So können Ressourcen aus verschiedenen Services dasselbe Tag zuweisen, um anzugeben, dass die Ressourcen verbunden sind. Sie können beispielsweise denselben Tag einem AWS Elemental MediaPackage-Channel und einem Endpunkt zuweisen, die Sie einer AWS Elemental MediaTailor-Konfiguration zuordnen.
- **Überwachen von AWS-Kosten.** You activate these tags on the AWS Billing and Cost Management dashboard. AWS uses the tags to categorize your costs and deliver a monthly cost allocation report to you. 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 erhalten Sie weitere Informationen zu Tags für AWS Elemental MediaTailor.

Unterstützte Ressourcen in AWS Elemental MediaTailor

Die folgenden Ressourcen in AWS Elemental MediaTailor unterstützen das Tagging:

- Kanäle
- Konfigurationen
- SourceLocations
- VodSources

Tag (Markierung)-Einschränkungen

Für Tags in AWS Elemental MediaTailor-Ressourcen gelten die folgenden grundsätzlichen Einschränkungen:

- 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 MediaTailorAPI-Referenz](#).

Problembhebung MediaTailor

Wenn Sie bei der Arbeit mit auf Wiedergabefehler oder ähnliche Schwierigkeiten stoßenAWS Elemental MediaTailor, lesen Sie die Themen in diesem Abschnitt.

Themen

- [Behebung von Wiedergabefehlern, die zurückgegeben wurden von MediaTailor](#)

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

Sie erhalten möglicherweise auch Fehler von der AWS Elemental MediaTailor-API bei Konfigurationsoperationen wie `PutPlaybackConfiguration` und `GetPlaybackConfiguration`. Informationen zu diesen Fehlertypen finden Sie in der [AWS Elemental MediaTailorAPI-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 Ausnahmenname und eine Fehlermeldung MediaTailor zurückgegeben.

AWS Elemental MediaTailor gibt zwei Klassen von Fehlern zurück:

- Clientfehler — 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

- [Client-Wiedergabefehler wurden zurückgegeben von AWS Elemental MediaTailor](#)
- [Serverwiedergabefehler wurden zurückgegeben von AWS Elemental MediaTailor](#)
- [Beispiele für Fehler bei der Wiedergabe](#)

Client-Wiedergabefehler 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 von abrufen AWS Elemental MediaTailor. Für die API ist die Ressource `GetPlaybackConfiguration/Name`. Einzelheiten finden Sie in der [AWS Elemental MediaTailor API-Referenz](#).

In der folgenden Tabelle werden die Client-Fehlercodes aufgelistet, die durch die Manifest-Manipulierungsaktivitäten von AWS Elemental MediaTailor zurückgegeben werden, sowie mögliche Ursachen und Maßnahmen, die Sie ergreifen können, um diese Probleme zu beheben.

| Code | Name der Ausnahme | Bedeutung | Vorgehensweise |
|------|---------------------|--|--|
| 40 | BadRequestException | 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 | 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 ManifestE |

| Code | Name der Ausnahme | Bedeutung | Vorgehensweise |
|------|-----------------------|--|--|
| | | ungültige Playback-Konfiguration oder Sitzungs-ID. | ndpointPrefix -Einstellung übereinstimmt, die von GetPlaybackConfiguration zurückgegeben wird. Wiederholen Sie Ihre Anforderung. |
| 40 | AccessDeniedException | Der in der Anforderung angegebene Host-Header stimmt nicht mit dem Präfix des Manifest-Endpunkts überein, das in der MediaTailor-Wiedergabe-URL konfiguriert ist. Ihr CDN ist möglicherweise falsch konfiguriert. | Überprüfen Sie Ihre CDN-Einstellungen und stellen Sie sicher, dass Sie das richtige 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. |

| Code | Name der Ausnahme | Bedeutung | Vorgehensweise |
|------|-------------------|---|---|
| 41 | Gone | Ein AWS Support-Mitarbeiter hat eine Spielsitzung 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](#).

Serverwiedergabefehler 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 von abrufen AWS Elemental MediaTailor. Für die API ist die Ressource `GetPlaybackConfiguration/Name`. Einzelheiten finden Sie in der [AWS Elemental MediaTailor API-Referenz](#).

In der folgenden Tabelle werden die Server-Fehlercodes aufgelistet, die von den Manifest-Verarbeitungsaktivitäten von AWS Elemental MediaTailor zurückgegeben werden, sowie mögliche Ursachen und Maßnahmen, die Sie ergreifen können, um diese Probleme zu beheben.

| Code | Name der Ausnahme | Bedeutung | Vorgehensweise |
|------|------------------------------|--|---|
| 50 | InternalServiceError | 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. |
| 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 Manifestformat 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 |

| Code | Name der Ausnahme | Bedeutung | Vorgehensweise |
|------|-------------------------|--|--|
| | | | 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 auch den gemeldeten Zustand von MediaTailor für Ihre AWS-Region 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” . |
| 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 Ursprungsservers und stellen Sie sicher, dass der Ursprungsserver innerhalb des Timeouts des Ursprungsservers 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 werden einige Beispiele für Wiedergabefehler aufgelistet, die möglicherweise in Befehlszeilen-Interaktionen mit AWS Elemental MediaTailor angezeigt werden.

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]}%
```

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 (Standardkontingentwert) | Beschreibung | |
|--|--|--|--|
| (Entscheidungsserver für Werbung, ADS) | | bis das Timeout bei einer offenen Verbindung zu einem Ad Decision Server (ADS) überschritten wird. Wenn bei einer Verbindung das Zeitlimit überschritten 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 personalisierte Manifeste bei serverseitiger Anzeigeneinblendung. Die Anzeigeneinfügung verarbeitet eingehende Anfragen nach Manifesten, Sitzungsinitialisierung, Tracking-Daten und Anzeigensegmenten. Dieses Kontingent ist einstellbar. | |

| Name | Default quota value (Standardkontingentwert) | Beschreibung | |
|---|--|---|--|
| Konfigurationen | 1.000 | Die maximale Anzahl von Konfigurationen, die MediaTailor zulässig ist. | |
| 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 Inhaltsursprungserver MediaTailor gewartet wird. Zeitüberschreitungen generieren HTTP 504 (GatewayTimeoutException) -Antwortfehler. | |

| Name | Default quota value (Standardkontingentwert) | 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. | |
| 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 (Standardkontingentwert) | Beschreibung | |
|---|--|--|--|
| Timeout für serverseitige 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 überschritten MediaTailor wird, der Beacon nicht ausgelöst werden kann und der Dienst eine <code>ERROR_FIRING_BEACON_FAILED</code> Meldung unter MediaTailor/AdDecisionServerInteractionsprotokolliert, 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 | |
|------------------------------|--|--|--|
| Kanal-Manifest-Anforderungen | 50 | Die maximale Anzahl von Manifest-Anforderungen pro Sekunde pro | |

| Name | Default quota value (Standardkontingentwert) | Beschreibung | |
|------------------|--|---|--|
| | | Channel-Assembly-Kanal. Hierbei handelt es sich um ein Kontingent auf Kontoebene. | |
| Kanal-Ausgaben | 5 | Die maximale Anzahl von Ausgaben pro Kanal. | |
| Kanäle pro Konto | 100 | Die maximale Anzahl von Kanälen pro Konto. | |
| 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-second Standard-Höchstlimit | Beschreibung | |
|-------------------------|--|---|--|
| ConfigureLogsForChannel | 1 | Konfigurieren Sie Protokolle für den Kanal. | |
| CreateChannel | 1 | Erstellen Sie einen Kanal. | |

| Name | transactions-per-second Standard-Höchstlimit | Beschreibung |
|----------------------|--|---------------------------------------|
| 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. |
| 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. |

| Name | transactions-per-second Standard-Höchstlimit | Beschreibung |
|----------------------------|--|--|
| 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. |
| 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 | Listet VOD-Quellen auf. |
| Verpackungskonfigurationen | 5 | Die maximale Anzahl von Paketkonfigurationen pro Quelle (ob Live- oder Video-on-Demand). |

| Name | transactions-per-second Standard-Höchstlimit | Beschreibung | |
|----------------------|--|---|--|
| PutChannelPolicy | 3 | Geben Sie eine Kanalrichtlinie ein. | |
| StartChannel | 1 | Starte einen Kanal. | |
| StopChannel | 1 | Stoppen Sie einen Kanal. | |
| TagResource | 1 | Kennzeichnen Sie eine Ressource. | |
| UntagResource | 1 | Enttaggen Sie eine Ressource. | |
| UpdateChannel | 1 | Aktualisiere einen Kanal. | |
| UpdateLiveSource | 1 | Aktualisiere eine Live-Quelle. | |
| 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-Standard-Dokument 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 Entwickler-Tools, SDKs, IDE-Tools und Befehlszeilen-Tools 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. Bietet außerdem Links zu hilfreichen Ressourcen wie z. B. Foren, technischen Fragen und Antworten, Übersicht zum Servicestatus 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 |
|---|--|-------------------|
| Variable aktualisiert | Geändert breakabilityStartTime zuavailabilityStartTime . | 6. Mai 2024 |
| Details zur Personalisierung hinzugefügt | Das Insertion Mode Personalisierungsdetail wurde hinzugefügt. | 6. Mai 2024 |
| Regeln des Programms | Neue Inhalte zu den Programmregeln hinzugefügt. | 25. April 2024 |
| Der Inhalt zum Erstellen von Kanälen wurde aktualisiert | 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 |
| 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 |

| | | |
|--|--|-------------------|
| <u>Die Manifest-Einstellungen wurden aktualisiert</u> | 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 <code>Daterange</code> im Vergleich zu <code>Scte35 Enhanced</code> Anzeigenmarkup-Typen hinzugefügt. | 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>VODQuelle 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 |

| | | |
|--|--|-------------------|
| Einrichtung MediaTailor und MediaPackage Workflows zur Verwendung von Live-Quellen | 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 |
| 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, der die Dekoration von Anzeigen-IDs in Manifesten beschreibt. | 24. April 2023 |
| Der EDGE Unterdrückungsmodus AFTER LIVE wurde hinzugefügt | AFTER_LIVE_EDGE Der Modus zur Anzeigenunterdrückung ist jetzt zusätzlich zum BEFORE_LIVE_EDGE Modus verfügbar. | 21. Februar 2023 |
| Neues As-Run-Protokoll | Neues Thema im As Run-Protokoll. | 19. Januar 2023 |
| IAMUpdates zu bewährten Verfahren | Aktualisierter Leitfaden zur Anpassung an die IAM bewährten Verfahren. Weitere Informationen finden Sie unter Bewährte Sicherheitsmethoden unter IAM . | 27. Dezember 2022 |

| | | |
|---|--|--------------------|
| IAMUpdates zu bewährten Methoden | Aktualisierter Leitfaden zur Anpassung an die IAM bewährten Verfahren. Weitere Informationen finden Sie unter Bewährte Sicherheitsmethoden unter IAM . | 27. Dezember 2022 |
| Inhalt der Kontingente aktualisiert | Die Informationen zu den Kontingenten wurden aktualisiert und neu organisiert. | 13. September 2022 |
| Kontingente für Kanalzusammenstellungen wurden hinzugefügt und korrigiert | Es wurden Kontingente für Live-Quellen, Konfigurationen für die Segmentzuweisung, Manifestanfragen und Kanaltransaktionen im MediaTailor Channel Assembly Service hinzugefügt. | 13. September 2022 |
| Kontingente für Channel-Assembly hinzugefügt und korrigiert | Es wurden Kontingente für Protokollierung, Kanal, Live-Quellen, Programme, Quelladressen und Kanalrichtlinien im MediaTailor Channel Assembly Service hinzugefügt. | 11. September 2022 |
| Neue Tabelle mit Warnmeldungen zur Kanalbaugruppe | Sie können jetzt eine Tabelle sehen, in der die Warnmeldungen zur Kanalmontage erklärt werden. | 01. September 2022 |
| Neue CloudWatch Amazon-Metriken | Neue CloudWatch Metriken hinzugefügt. | 26. Juni 2022 |
| Anzeige ruft Themen auf | Link zur Prefetch-Unterstützung für VAST Antworten hinzugefügt. | 25. Mai 2022 |

[Neue Anforderungsvariablen ADS](#)

MediaTailor unterstützt jetzt diese zusätzlichen SCTE -35 Variablen in ADS Anfragen:

```
scte.segmentation_
type_id ,scte.avai
ls_expected ,scte.deli
very_not_restricte
d_flag ,scte.segm
ent_num ,scte.sub_
segment_n
um ,scte.segments_expe
cted ,scte.sub_
segments_expected ,
scte.device_restri
ctions scte.no_r
egional_blackout_f
lag ,scte.arch
ive_allowed_flag
undscte.segmentation_
event_id .
```

27. April 2022

[Neues Thema für IAM verwaltete Richtlinien](#)

Zwei neue verwaltete Richtlinien für hinzugefügt MediaTailor.

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 AWSElementalMediaTailorFull Access 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 Nachrichten 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 |

| | | |
|--|--|----------------|
| <u>Neue Informationen zum Quellentyp</u> | Für Standardkanäle wurden Informationen über die Art der Quelle hinzugefügt, die das Programm wiedergibt. | 13. Juni 2021 |
| <u>Neue Dokumentation MediaTailor zu Live-Quellen</u> | 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 |
| <u>Support für zusätzliche UPID Typen</u> | MediaTailor unterstützt jetzt die UPID Segmentierungstypen ADS Information (0xE) und Benutzerdefiniert (0x1). | 15. April 2021 |
| <u>Neue dynamische Segmentierungsvariablen UPID</u> | 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 MPU UPID Segmentierungstyp (0xC) für Podbuster-Workflows verwendet. | 15. April 2021 |
| <u>Neue Beschreibung des Channel-Assembly-Services</u> | 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 Kanalmontage wurden 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 Werbermarkierungen 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 Log-Modus. | 14. August 2020 |
| <u>Klarstellung zum EXT OUT Dauerattribut -X- CUE für Stoßstangen</u> | Die Anforderungen für Bumpers wurden aktualisiert, sodass 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 verfügbar für DASH</u> | Die Anzeigenunterdrückung ist jetzt verfügbar für DASH. Die Einschränkung „HLSNur“ wurde aus dem Thema Anzeigenunterdrückung entfernt. | 3. Juni 2020 |

| | | |
|--|---|-------------------|
| <u>Konsolenspezifische Namen aktualisieren</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 Berichterstattung 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>DASHVODManifestiert</u> | Unterstützung für Video-on-Demand (VOD) DASH - Manifeste vom Ursprungserver mit Manifestausgabe für mehrere Perioden 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> | Die Grenzwerte für ADS Weiterleitungen und ADS Timeouts wurden aktualisiert. | 18. Dezember 2019 |


| | | |
|--|---|--------------------|
| CDNbewährte Verfahren | Es wurde ein Abschnitt über bewährte Methoden für personalisierte Manifeste im Content Distribution Network (CDN) hinzugefügt. | 13. Dezember 2019 |
| Dokumentieren Sie das Verhalten beim Live-Pre-Roll | 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>DASHManifeste für einzelne Perioden</u> | Unterstützung für einperiodische DASH Manifeste vom Ursprungsserver mit Ausgabe von Manifesten für mehrere Perioden hinzugefügt. | 4. April 2019 |
| <u>Support für SCTE -35 UPIDs in der ADS URL</u> | Unterstützung für das Einfügen einer eindeutigen Programm-ID (UPID) in den Anzeigenentscheidungsserver (ADS) URL wurde hinzugefügt. Dies ermöglicht die ADS Bereitstellung von zielgerichteter Werbung auf Programmebene innerhalb eines linearen Live-Streams. | 28. März 2019 |
| <u>Die clientseitige Berichterstattung unterstützt Begleitanzeigen</u> | Für die clientseitige Berichterstattung umfasst die AWS Elemental MediaTailor URL Tracking-Antwort jetzt Metadaten für Begleitanzeigen. | 28. März 2019 |
| <u>HLSDokumentation zur Werbemarkierung</u> | Es wurde ein Abschnitt hinzugefügt, in dem unterstützte HLS Werbemarkierungen beschrieben werden. | 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 zur Verwendung CloudTrail zum Protokollieren von Aktionen in der hinzugefügt AWS Elemental MediaTailor API. | 11. Februar 2019 |
| Abschnitt zu Wiedergabefehlern hinzugefügt | Es wurden Informationen zu den Fehlern hinzugefügt, die MediaTailor bei der Wiedergabe als Antwort auf Anfragen von einem Player oder einem Content Delivery Network auftreten können (CDN). | 4. Februar 2019 |
| DASHBase64-codierte Binärdatei | Es wurde Unterstützung für die Bereitstellung von Splicing-Informationen in Manifesten im base64-kodierten binären Format innerhalb von <code><scte35:Signal></code> - und <code><scte35:Binary></code> -Markern hinzugefügt. | 4. Januar 2019 |

| | | |
|---|---|-------------------|
| DASHZeitsignal | Es wurde Unterstützung für die Bereitstellung von Splicing-Informationen in Manifesten innerhalb von <code><scte35:TimeSignal></code> -Markern hinzugefügt. | 5. Dezember 2018 |
| DASHUnterstützung für den Standort | Unterstützung für das MPEG DASH <code><Location></code> -Tag hinzugefügt. | 4. Dezember 2018 |
| DASHUnterstützung | 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 Messwerte für Ad Decision Server (ADS) und Origin Timeouts hinzugefügt und die Definitionen für Fehler ADS und Herkunft aktualisiert, sodass auch Antworten mit Timeout enthalten sind. | 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 FRA Regionen PDX und hinzugefügt. | 18. Juli 2018 |

| | | |
|---|---|-------------------|
| VAST/VPAID | Es wurden Informationen über VAST und hinzugefügtVPAID. | 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 |
| IAMInformationen zur Politik | Es wurden 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, bei Navigations- oder Kommunikationssystemen, bei der Flugsicherung oder bei 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.