



Manual do usuário

AWS Elemental MediaTailor



AWS Elemental MediaTailor: Manual do usuário

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

As marcas comerciais e imagens de marcas da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestige a Amazon. Todas as outras marcas comerciais que não pertencem à Amazon pertencem a seus respectivos proprietários, que podem ou não ser afiliados, patrocinados pela Amazon ou ter conexão com ela.

Table of Contents

O que é AWS Elemental MediaTailor?	1
Requisitos do servidor de origem	1
MediaTailor conceitos	2
Tipos de playlist HLS	2
Tipos de manifesto DASH	4
Conceitos de inserção de anúncios	10
Conceitos de montagem de canais	11
Como funciona a inserção de anúncios	12
Fluxo de eventos de inserção de anúncios	14
Serviços relacionados	17
Acessando MediaTailor	18
Preços	18
Regiões	18
Configurar	19
Inscreva-se para um Conta da AWS	19
Começando com MediaTailor	20
Introdução à inserção MediaTailor de anúncios	20
Pré-requisitos	21
Etapa 1: Acesso MediaTailor	21
Etapa 2: preparar um stream	21
Etapa 3: Configurar o URL de solicitação do ADS e os parâmetros de consulta	24
Etapa 4: Criar uma configuração	25
Etapa 5: testar a configuração	27
Etapa 6: enviar a solicitação de reprodução para MediaTailor	28
Etapa 7 (opcional): monitorar a atividade	29
Etapa 8: Limpeza	30
Introdução à montagem MediaTailor de canais	31
Pré-requisitos	32
Etapa 1: criar um local de origem	32
Etapa 2: Adicionar fontes de VOD ao seu local de origem	33
Etapa 3: criar um canal	34
Etapa 4: adicionar programas à programação do seu canal	36
Etapa 5 (optional): Use MediaTailor para inserir anúncios personalizados em seu stream	37
Etapa 6: inicie seu canal	38

Etapa 7: teste seu canal	38
Etapa 8: Limpeza	39
Inserindo anúncios	40
Codecs de áudio e vídeo compatíveis	41
Compreender o comportamento de inserção de anúncios	41
Comportamento de costura de anúncios para VOD	41
Comportamento de costura de anúncios ao vivo	44
Visão geral da inserção de anúncios guiada pelo servidor	47
Ativar na configuração de reprodução	48
Crie uma sessão guiada pelo servidor	48
Rastreamento de anúncios com o SGAI	49
Pré-busca guiada com batimento cardíaco manifesto	50
Compatibilidade de recursos SGAI	51
Configuração ao vivo do SGAI	53
Configuração SGAI VOD	57
Requisitos de integração do servidor de anúncios	61
Requisitos VAST	62
Requisitos do VPAID	64
Gerenciamento de configuração	65
Criar uma configuração	65
Visualizando detalhes da configuração	74
Editando as configurações	74
Excluindo configurações	74
Integrando uma fonte de conteúdo	75
Requisitos do arquivo de origem	75
Integrando uma fonte HLS	76
Integrando uma fonte MPEG-DASH	88
Protegendo as interações de origem com o SigV4	107
Integração com as plataformas de anúncios do Google	111
Server-side integração	112
Client-side integração	114
Personalizando o comportamento de anúncios com a supressão de anúncios	115
Configurando a supressão de quebras de anúncios	115
Amortecedor e inserção	129
Configurando amortecedores	130
Pré-rolô e inserção	131

Inserção de anúncios em ardósia	132
Configurando a ardósia	133
Como funciona o limite de personalização	134
Configuração do Slate e VPAID	137
Transcodificação de anúncios	137
Transcodificação dinâmica	138
Perfis de transcodificação personalizados	140
Pré-busca de anúncios	144
Como funciona a pré-busca	145
Criação de agendas de pré-busca	150
TPS-based modelagem de tráfego	157
Excluindo agendas de pré-busca	158
Anúncios pré-condicionados	159
Requisitos de anúncios pré-condicionados	160
Fluxo de trabalho de anúncios pré-condicionados	163
Funções de monetização	164
Como as funções se encaixam no MediaTailor fluxo de trabalho	165
Guia de início rápido	166
Hooks do ciclo de vida	170
Tipos de funções e composição	179
Criando e gerenciando	189
Referência de expressão JSonata	192
Exemplos de funções	203
Solução de problemas e monitoramento	208
Limites	217
Variáveis de anúncios dinâmicos	220
Variáveis de sessão	222
Variáveis do jogador	235
Variáveis de domínio	237
Aliases de configuração	238
Passando parâmetros do ADS	246
Roteamento de parâmetros	250
Integração do MediaPackage	251
Comportamento da sessão	253
Referência a parâmetros	255
Solução de problemas de parâmetros	257

Solução de problemas de aliases	261
MediaTailor parâmetros de consulta do manifesto	265
Parâmetros de consulta de origem	265
MediaTailorrestrições de caracteres	266
MediaTailorSessões implícitas do HLS	268
MediaTailor Sessões implícitas do DASH	269
MediaTailor inicialização explícita da sessão	271
MediaTailorcomportamento específico do protocolo	273
MediaTailor Integração CDN	276
Dados de relatórios e rastreamento	277
Server-side rastreamento	278
Client-side rastreamento	295
Anúncios de sobreposição	372
Pré-requisitos para usar anúncios sobrepostos	373
Introdução	374
Registro e métricas	387
Cobrança por anúncios sobrepostos em MediaTailor	390
Decoração de ID do anúncio	390
Estado da sessão	391
Manifestos e inserção de metadados de anúncios	393
Interações do Ad Decision Server (ADS)	417
API de rastreamento do lado do cliente	418
Criação de fluxos montados lineares	420
Trabalhando com locais de origem	420
Criação de um local de origem	421
Configurando a autenticação para seu local de origem	423
Trabalhando com fontes de VOD	432
Trabalhando com fontes ao vivo	436
Usando configurações de pacotes	441
Cache de manifestos	442
Como trabalhar com canais	442
Criar um canal	442
Usando grupos de origem com as saídas do seu canal	445
Excluir um canal	446
Adicionando um programa	446
Criando um programa	446

Definição de coortes de público e conteúdo alternativo	455
Gerando manifestos específicos para o público	457
Inserir anúncios e quebras publicitárias	458
Configurar a inserção de anúncios	458
SCTE-35 mensagens para pausas publicitárias	460
Mudando o horário da reprodução de um canal	466
Usando uma CDN	468
Seleção de CDN	469
Planeje a integração com a CDN	471
Estime os requisitos de tráfego	472
Configurar estratégias de otimização	473
Personalize o planejamento	475
Configure o monitoramento e o escalonamento	477
Otimize custos	479
Teste sua implementação	481
Solução de problemas comuns	483
Configurar a integração com a CDN	485
Componentes e requisitos de integração de CDN	486
Pré-requisitos para integração com CDN	21
Etapas de configuração da integração CDN	487
Cabeçalhos necessários para integração com MediaTailor CDN	487
Configurar comportamentos de roteamento de CDN	489
Configurar o mapeamento CDN	494
Práticas recomendadas de segurança	497
Inserção de anúncios com CDN	499
O que você precisará	499
Antes de começar	500
Benefícios da integração com a CDN	500
Entenda a arquitetura CDN	501
Configurar a inserção básica de anúncios	508
SSAI com montagem de canais	511
Otimize o desempenho da CDN	519
Monitore as operações de CDN	521
Solucione problemas de inserção de anúncios com CDNs	527
Montagem de canais com CDN	531
O que você precisará	532

Antes de começar	533
Benefícios da integração com a CDN	533
Entenda a arquitetura CDN	534
Configuração básica	537
Configurar base URLs	537
Implementar inserção de anúncios	538
Configurar a visualização com mudança de horário	541
Monitore as operações de CDN	543
Guia completo de otimização	543
MediaPackage Integração CDN	544
Entendendo o fluxo de MediaPackage trabalho e o CDN	544
Pré-requisitos	545
Configuração de configurações essenciais	545
Verificação de integração	548
Próximas etapas	549
Filtragem de manifestos	550
Cache CDN	556
Desempenho da CDN	561
Solução de problemas de integração de CDN	566
Integração do CloudFront	571
CloudFront Configuração básica	571
Otimização de desempenho	578
Resiliência multirregional	582
Monitoramento e solução de problemas	586
Configuração de CDN de terceiros	591
Pré-requisitos	592
Configurar regras de armazenamento em cache da CDN	593
Verifique sua configuração de CDN	596
Otimize o desempenho da CDN	597
Solucionar problemas de CDN de terceiros	598
Otimização do desempenho da CDN	600
Otimização do cache	600
Otimização do roteamento de solicitações	606
Benchmarks de desempenho	606
Otimização avançada	607
Monitoramento de CDN	610

Métricas essenciais de desempenho da CDN	611
Configurar ferramentas de monitoramento	613
Lista de verificação de implementação	615
Tópicos relacionados	615
Teste de integração CDN	616
Pré-requisitos de teste	616
Metodologia de teste sistem	617
Lista de verificação de pré-implantação	622
Referência de ferramentas de teste	623
Solucionar problemas de integração com a CDN	624
Lista de verificação de diagnóstico rápido	625
Teste de integração	627
Erros manifestos 404	631
Problemas de entrega do manifesto	632
Problemas de entrega do segmento	634
Problemas de gerenciamento de sessões	636
Problemas de cronometragem de intervalos publicitários	638
Problemas de desempenho e latência	641
Comportamento inconsistente do dispositivo	648
Preparação para solução de	648
Guias de solução de problemas específicos do fluxo de trabalho	651
Referência de análise de log de integração CDN	652
Interpretação do log CDN	653
Referência de códigos de erro	655
Ferramentas de análise de registros	657
MediaTailor recursos de configuração de log	658
CloudFormation Automation	659
Por que usar AWS CloudFormation	660
Preparar-se para implantação	661
Implantação do modelo do	664
Use os recursos implantados	665
Teste e valide sua implantação	668
Solução de problemas de implantação do	669
Personalize o modelo	671
Referência do modelo	674
CloudFrontConfiguração de produção	680

Three-origin arquitetura	680
Precedência e configuração do comportamento do cache	682
Obtenha suporte para integração de CDN	689
Quando escalar para o Support	689
Coletar informações para fins de suporte	690
Recursos de suporte	691
Práticas recomendadas de casos de suporte	691
Segurança	693
Proteção de dados	694
Criptografia de dados	695
Gerenciamento de Identidade e Acesso	695
Público	696
Autenticação com identidades	696
Gerenciar o acesso usando políticas	697
Como AWS Elemental MediaTailor funciona com o IAM	699
Identity-based exemplos de políticas	704
Resource-based exemplos de políticas	707
AWS políticas gerenciadas	708
Uso de perfis vinculados ao serviço	710
Solução de problemas de identidade e acesso	713
Validação de conformidade	716
Resiliência	717
Segurança da infraestrutura	717
Cross-service prevenção delegada confusa	718
Registro em log e monitoramento	718
CloudWatch Alarmes	719
CloudTrail troncos	719
AWS Trusted Advisor	719
Monitorar e atribuir tags (tagging)	720
Visualizar logs	721
Registros do ADS	721
Registros de manifestos	782
Registros de transcodificação	787
Usando registros vendidos	791
Gravando registros em CloudWatch registros	795
Controle do volume dos registros de sessão de inserção de anúncios	804

Filtrando registros e eventos	807
Gerando registros de depuração	810
Monitoramento com CloudWatch métricas	815
AWS Elemental MediaTailor CloudWatch métricas	816
AWS Elemental MediaTailor CloudWatch dimensões	833
Usando métricas para diagnosticar manifestos obsoletos	834
Gravação de chamadas de API	837
AWS Elemental MediaTailor informações em CloudTrail	838
Noções básicas AWS Elemental MediaTailor entradas do arquivo de log	839
Recebendo alertas de montagem de canais	841
Solução de problemas de alertas relacionados ao manifest	846
Visualizando alertas	846
Gerenciamento de alertas	847
Tagging resources	847
Recursos compatíveis	848
Restrições de tags	848
Como gerenciar tags	848
Monitor de fluxo de trabalho	849
Componentes do monitor de fluxo de trabalho	850
Serviços com suporte	851
Configurar o monitor de fluxo de trabalho	852
Usar o monitor de fluxo de trabalho	871
Dual-stack endpoints	874
O que são endpoints de pilha dupla?	874
Recursos compatíveis	874
Dual-stack formato de endpoint	874
Usar endpoints de pilha dupla	875
Console de gerenciamento da AWS	875
AWS Command Line Interface	875
AWS SDK	875
Considerações	876
Cotas	877
Cotas na inserção de anúncios	877
Cotas na montagem do canal	883
Solução de problemas	888
Solução de problemas de fluxo de eventos	888

Fluxos incompletos	888
Análise de temporização	891
Problemas comuns	892
Práticas recomendadas de monitoramento	896
Recursos adicionais do	897
Solução de problemas de reprodução	898
Erros de cliente	898
Erros de servidor	900
Exemplos	902
Desduplicação de beacons	903
Solução de problemas de parâmetros	904
Solução de problemas com o ignorar anúncios	904
Sintomas	905
Ignorar motivos	905
NEW_CREATIVE pulando	906
Ignorando o tempo limite do ADS	908
Ignorando a análise VAST	909
Ignorando a incompatibilidade de duração	910
Ignorando a variável da sessão	911
Monitorando e pulando	912
Melhores práticas de prevenção	913
Materiais de referência	914
MediaTailor recursos	918
Histórico do documento	920
.....	cmxlii

O que é AWS Elemental MediaTailor?

AWS Elemental MediaTailor é um serviço escalável de inserção de anúncios e montagem de canais executado na AWS nuvem. Com isso MediaTailor, você pode veicular conteúdo publicitário direcionado aos espectadores e criar fluxos lineares, mantendo a qualidade da transmissão em aplicativos de vídeo over-the-top (OTT). MediaTailor A inserção de anúncios é compatível com Apple HTTP Live Streaming (HLS) e MPEG Dynamic Adaptive Streaming over HTTP (DASH) para vídeos sob demanda (VOD) e fluxos de trabalho ao vivo.

AWS Elemental MediaTailor a inserção de anúncios oferece avanços importantes em relação aos sistemas tradicionais de rastreamento de anúncios: os anúncios são mais monetizados, mais consistentes em qualidade e resolução de vídeo e mais fáceis de gerenciar em ambientes multiplataforma. MediaTailor simplifica seu fluxo de trabalho de anúncios ao permitir que todos os IP-connected dispositivos renderizem anúncios da mesma forma que renderizam outros conteúdos. O serviço também oferece rastreamento avançado de exibições de anúncios, o que aumenta ainda mais a monetização do conteúdo.

AWS Elemental MediaTailor a montagem de canais é um serviço somente de manifesto que permite criar canais de streaming linear usando seu conteúdo de vídeo sob demanda (VOD) existente. MediaTailor nunca toca em seus segmentos de conteúdo, que são servidos diretamente do seu servidor de origem. Em vez disso, MediaTailor busca os manifestos de sua origem e os usa para montar uma janela de manifesto deslizante ativa que faz referência aos segmentos de conteúdo subjacentes.

MediaTailor A montagem de canais facilita a monetização de seu canal inserindo intervalos de anúncios em seu stream sem precisar condicioná-lo com marcadores. SCTE-35 Você pode usar a montagem de canais com inserção de MediaTailor anúncios ou outro serviço de inserção de anúncios do lado do servidor.

Requisitos do servidor de origem

AWS Elemental MediaTailor tem requisitos específicos para comunicação com o servidor de origem:

- Portas suportadas - MediaTailor só aceita origens usando portas HTTP e HTTPS padrão:
 - Porta 80 para conexões HTTP
 - Porta 443 para conexões HTTPS

MediaTailor não oferece suporte a portas personalizadas para comunicação com o servidor de origem.

- Requisitos de protocolo - Para comunicação segura, MediaTailor requer HTTPS para determinados tipos de origem e cenários de autenticação. Para obter mais informações, consulte [the section called “Integrando uma fonte de conteúdo”](#).

MediaTailor conceitos

Aqui está uma visão geral dos conceitos usados em todo o Guia AWS Elemental MediaTailor do Usuário.

Tipos de playlist HLS

O HTTP Live Streaming (HLS) usa dois tipos principais de playlists: playlists multivariantes e playlists de mídia. Entender a diferença entre esses tipos de playlist é essencial para configurar e solucionar problemas de MediaTailor fluxos de trabalho.

Lista de reprodução multivariante

Uma playlist multivariante é o arquivo de índice de nível superior em um fluxo de trabalho de streaming HLS que lista todas as representações disponíveis do conteúdo. Ele contém referências a playlists de mídia, mas não contém nenhum segmento de mídia em si. A playlist multivariante permite que os jogadores selecionem a reprodução mais adequada com base nas condições da rede, nos recursos do dispositivo ou nas preferências do usuário.

Esse tipo de playlist também é conhecido por vários outros nomes em vários contextos:

- Playlist principal (termo obsoleto)
- Manifesto principal (termo comum da indústria)
- Playlist principal
- Playlist principal
- Arquivo de índice
- Mestre M3U8

Em MediaTailor fluxos de trabalho, a playlist multivariante é o ponto de entrada para solicitações de reprodução e é onde a personalização de anúncios começa.

Example Exemplo de playlist multivariante

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-INDEPENDENT-SEGMENTS
#EXT-X-STREAM-INF:BANDWIDTH=2665726,AVERAGE-
BANDWIDTH=2526299,RESOLUTION=960x540,CODECS="avc1.640029,mp4a.40.2",SUBTITLES="subtitles"
index_1.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=4786126,AVERAGE-
BANDWIDTH=4547599,RESOLUTION=1280x720,CODECS="avc1.640029,mp4a.40.2",SUBTITLES="subtitles"
index_2.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=8171787,AVERAGE-
BANDWIDTH=7768157,RESOLUTION=1920x1080,CODECS="avc1.640029,mp4a.40.2",SUBTITLES="subtitles"
index_3.m3u8
```

Playlist de mídia

Uma playlist de mídia contém as informações reais do segmento de mídia para uma representação específica (nível de qualidade) do conteúdo. Ele inclui informações de tempo URLs, segmentos e outros metadados necessários para a reprodução de uma única representação. Cada playlist de mídia representa uma combinação diferente de resolução, taxa de bits e outras características.

Esse tipo de playlist também é conhecido por vários outros nomes:

- Playlist variante
- Manifesto infantil (termo comum da indústria)
- Lista de fragmentos
- Mídia M3U8
- Playlist de renderização

Nos MediaTailor fluxos de trabalho, as playlists de mídia são personalizadas para incluir segmentos de conteúdo e segmentos de anúncios na sequência correta.

Example Exemplo de playlist de mídia

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:6
#EXT-X-MEDIA-SEQUENCE:123456
#EXT-X-DISCONTINUITY-SEQUENCE:0
```

```
#EXTINF:6.0,  
segment123456.ts  
#EXTINF:6.0,  
segment123457.ts  
#EXT-X-DISCONTINUITY  
#EXTINF:5.0,  
ad-segment1.ts  
#EXTINF:5.0,  
ad-segment2.ts  
#EXT-X-DISCONTINUITY  
#EXTINF:6.0,  
segment123458.ts
```

Note

Tanto as playlists multivariantes quanto as de mídia são arquivos baseados em texto que usam a extensão. `.m3u8` A terminologia desses arquivos varia em diferentes documentações e contextos, mas as funções fundamentais permanecem as mesmas na arquitetura de streaming do HLS.

Ao configurar MediaTailor, você normalmente fornece o URL para a playlist multivariante na origem do seu conteúdo. MediaTailor em seguida, gerencia a personalização da playlist multivariante e das playlists de mídia para inserir anúncios de acordo com sua configuração.

Para obter mais informações sobre as especificações da playlist HLS, consulte o [HTTP Live Streaming RFC](#).

Tipos de manifesto DASH

O Dynamic Adaptive Streaming over HTTP (DASH) usa um manifesto Media Presentation Description (MPD) para fornecer conteúdo de streaming. Compreender a estrutura e os componentes dos manifestos do DASH é essencial para configurar e solucionar problemas de fluxos de trabalho. MediaTailor

MPD (Descrição da apresentação de mídia)

O MPD é o arquivo de manifesto principal no streaming do DASH que descreve a estrutura e a disponibilidade do conteúdo de mídia. Ele contém informações sobre períodos, conjuntos de adaptação, representações e segmentos que compõem o conteúdo de streaming.

Esse tipo de manifesto também é conhecido por vários outros nomes em vários contextos:

- Manifesto DASH
- DASH MOD
- Manifesto principal (quando comparado ao HLS)
- manifesto de apresentação

Em MediaTailor fluxos de trabalho, o MPD é o ponto de entrada para solicitações de reprodução e é onde a personalização de anúncios começa.

Example Exemplo de manifesto MPD

```
<?xml version="1.0" encoding="UTF-8"?>
<MPD xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011
DASH-MPD.xsd" profiles="urn:mpeg:dash:profile:isoff-live:2011"
type="dynamic" minBufferTime="PT5.000S" maxSegmentDuration="PT2.005S"
availabilityStartTime="2020-01-01T00:00:00Z" publishTime="2020-01-01T12:30:00Z"
minimumUpdatePeriod="PT2.000S" timeShiftBufferDepth="PT5M">
  <Period id="1" start="PT0.000S">
    <AdaptationSet id="1" contentType="video" segmentAlignment="true"
    bitstreamSwitching="true" frameRate="30000/1001" maxWidth="1920" maxHeight="1080"
    par="16:9">
      <Representation id="1" mimeType="video/mp4" codecs="avc1.640028" width="1920"
      height="1080" bandwidth="5000000">
        <SegmentTemplate timescale="90000" initialization="init-stream
$RepresentationID$.m4s" media="chunk-stream$RepresentationID$-$Number%05d$.m4s"
startNumber="1" duration="180000"/>
      </Representation>
      <Representation id="2" mimeType="video/mp4" codecs="avc1.4d401f" width="1280"
      height="720" bandwidth="2800000">
        <SegmentTemplate timescale="90000" initialization="init-stream
$RepresentationID$.m4s" media="chunk-stream$RepresentationID$-$Number%05d$.m4s"
startNumber="1" duration="180000"/>
      </Representation>
      <Representation id="3" mimeType="video/mp4" codecs="avc1.4d401e" width="640"
      height="360" bandwidth="1100000">
        <SegmentTemplate timescale="90000" initialization="init-stream
$RepresentationID$.m4s" media="chunk-stream$RepresentationID$-$Number%05d$.m4s"
startNumber="1" duration="180000"/>
      </Representation>
    </AdaptationSet>
  </Period>
</MPD>
```

```

<AdaptationSet id="2" contentType="audio" segmentAlignment="true" lang="eng">
  <Representation id="4" mimeType="audio/mp4" codecs="mp4a.40.2"
audioSamplingRate="48000" bandwidth="128000">
    <SegmentTemplate timescale="48000" initialization="init-stream
$RepresentationID$.m4s" media="chunk-stream$RepresentationID$-$Number%05d$.m4s"
startNumber="1" duration="96000"/>
  </Representation>
</AdaptationSet>
</Period>
</MPD>

```

Período

Um período é uma seção temporal de uma apresentação do DASH. Cada período contém um ou mais conjuntos de adaptação e representa um período de tempo de mídia. Em fluxos de trabalho de inserção de anúncios, períodos separados são normalmente usados para delimitar entre conteúdo e anúncios.

Esse componente também é conhecido por vários outros nomes:

- Segmento de conteúdo
- Seção de cronograma
- Segmento do programa

Nos MediaTailor fluxos de trabalho, os períodos são usados para separar o conteúdo principal do conteúdo do anúncio, com cada anúncio normalmente representado por seu próprio período.

Exemplo Exemplo de período

```

<Period id="ad-period-1" start="PT30.000S" duration="PT15.000S">
  <AdaptationSet id="1" contentType="video" segmentAlignment="true"
bitstreamSwitching="true" frameRate="30000/1001" maxWidth="1920" maxHeight="1080"
par="16:9">
    <Representation id="1" mimeType="video/mp4" codecs="avc1.640028" width="1920"
height="1080" bandwidth="5000000">
      <SegmentTemplate timescale="90000" initialization="ad1/init-stream
$RepresentationID$.m4s" media="ad1/chunk-stream$RepresentationID$-$Number%05d$.m4s"
startNumber="1" duration="180000"/>
    </Representation>
  </AdaptationSet>
</Period>

```

AdaptationSet

E AdaptationSet agrupa um conjunto de versões codificadas intercambiáveis de um ou vários componentes de conteúdo de mídia. Por exemplo, um AdaptationSet pode conter vários níveis de qualidade de vídeo, enquanto outro pode conter várias opções de idioma de áudio.

Esse componente também é conhecido como:

- Grupo de componentes de mídia
- Conjunto de transmissões
- Grupo de faixas

Nos MediaTailor fluxos de trabalho, AdaptationSets são preservados durante a inserção do anúncio para manter os tipos de mídia consistentes entre o conteúdo e os anúncios.

Representação

Uma representação é uma versão codificada específica do conteúdo de mídia em um AdaptationSet. Cada representação normalmente difere em taxa de bits, resolução ou outros parâmetros de codificação, permitindo que os clientes selecionem a versão mais apropriada com base nas condições da rede e nos recursos do dispositivo.

Esse componente também é conhecido como:

- Rendição (semelhante ao HLS)
- Nível de qualidade
- Variante de taxa de bits
- Variante de transmissão

Em MediaTailor fluxos de trabalho, as Representações em Períodos de Anúncios são combinadas da forma mais próxima possível com as Representações em Períodos de Conteúdo para garantir uma experiência de visualização tranquila.

Segmento

Um segmento é uma unidade de dados de mídia que pode ser referenciada individualmente por um URL. Os segmentos contêm o conteúdo real da mídia (vídeo, áudio etc.) e são referenciados no MPD. Há dois tipos principais de segmentos no DASH:

- Segmento de inicialização: contém informações de inicialização para uma representação, como parâmetros de codec e informações de tempo.

- Segmento de mídia: contém os dados reais de mídia para um intervalo de tempo específico em uma representação.

Em MediaTailor fluxos de trabalho, os segmentos geralmente URLs são modificados para apontar para o conteúdo apropriado ou arquivos de mídia publicitária.

Note

Os manifestos do DASH são arquivos baseados em XML que normalmente usam a extensão. .mpd A terminologia desses arquivos e seus componentes pode variar em diferentes documentações e contextos, mas a estrutura fundamental permanece a mesma na arquitetura de streaming do DASH.

Ao configurar MediaTailor, você fornece o URL para o manifesto MPD em sua origem de conteúdo. MediaTailor em seguida, gerencia a personalização do manifesto, normalmente inserindo períodos adicionais para anúncios de acordo com sua configuração.

Para obter mais informações sobre as especificações do manifesto do DASH, consulte as Diretrizes de [implementação do DASH Industry Forum](#).

Conceitos avançados do manifesto DASH

Ao trabalhar com manifestos do DASH em MediaTailor, entender os seguintes conceitos avançados pode ajudá-lo a configurar e solucionar problemas de seus fluxos de trabalho de streaming com mais eficiência:

Cálculo da hora de início do período

Nos manifestos do DASH, os horários de início do período são calculados com base no `availabilityStartTime` atributo do MPD e no atributo do período. `start` Para transmissões ao vivo, a hora de início é relativa à hora de início da disponibilidade, enquanto para conteúdo VOD, geralmente é relativa ao início da apresentação.

Ao MediaTailor inserir períodos de anúncios, ele calcula cuidadosamente os horários de início para garantir transições perfeitas entre conteúdo e anúncios. Esse cálculo leva em consideração:

- A hora de início do Período original
- A duração dos períodos de anúncios anteriores

- Qualquer diferença de tempo especificada na resposta do servidor de decisão do anúncio

Cálculos de tempo de pré-lançamento

Os anúncios pré-lançados nos manifestos do DASH exigem tratamento especial porque aparecem antes do início do conteúdo principal. MediaTailor insere anúncios pré-lançados como Períodos separados no início do manifesto com horários de início e durações apropriados.

Para anúncios pré-cadastrados,: MediaTailor

- Cria um novo período para cada anúncio pré-lançado
- Define a hora de início do primeiro período de pré-inscrição como 0
- Ajusta a hora de início do Período do conteúdo principal para considerar a duração total de todos os anúncios pré-lançados

Live-to-VOD transições

Os manifestos do DASH podem passar do formato ao vivo para o formato VOD (Video on Demand) quando uma transmissão ao vivo termina. Essa transição envolve a alteração do type atributo do MPD de dynamic para static e o ajuste de outros atributos, como e. timeShiftBufferDepth minimumUpdatePeriod

Quando MediaTailor os processos se manifestam durante live-to-VOD as transições, ele garante que:

- Os marcadores de anúncios são preservados no manifesto VOD
- Os horários de início do período são ajustados para manter o tempo adequado
- O manifesto permanece compatível com clientes de reprodução de VOD

Manipulação de DRM

As informações de gerenciamento de direitos digitais (DRM) nos manifestos do DASH geralmente são incluídas nos ContentProtection elementos ou nas representações. AdaptationSets MediaTailor preserva esses elementos durante a inserção do anúncio para garantir que a proteção do conteúdo permaneça intacta.

Ao processar conteúdo protegido por DRM,: MediaTailor

- Mantém todos os ContentProtection elementos do manifesto original
- Garante que o conteúdo do anúncio use esquemas de DRM compatíveis, se aplicável
- Preserva todos os atributos e elementos relacionados ao DRM em todo o manifesto

Bandeiras SCC

Os sinalizadores de Controle de Conteúdo Suplementar (SCC) nos manifestos do DASH fornecem informações adicionais sobre as características do conteúdo e os requisitos de reprodução. Esses sinalizadores são normalmente incluídos como atributos ou elementos na estrutura do MPD.

Os sinalizadores comuns do SCC que MediaTailor processam incluem:

- Informações de classificação de conteúdo
- Recursos de acessibilidade (legendas ocultas, descrições de áudio)
- Avisos consultivos de conteúdo
- Restrições de reprodução

MediaTailor preserva esses sinalizadores durante o processamento do manifesto para garantir que todos os metadados do conteúdo sejam mantidos no manifesto personalizado.

A compreensão desses conceitos avançados ajuda você a configurar o desempenho MediaTailor ideal e solucionar quaisquer problemas que possam surgir em seus fluxos de trabalho de streaming do DASH.

Conceitos de inserção de anúncios

Aqui está uma visão geral dos conceitos relacionados à inserção de anúncios.

Disponibilidade do anúncio

Uma unidade específica de tempo publicitário em um intervalo publicitário que pode ser vendida aos anunciantes. Um intervalo de anúncio pode conter vários anúncios disponíveis. Quando MediaTailor recebe uma resposta VAST de um servidor de decisão de anúncios, ele preenche essas informações com anúncios personalizados.

Pausa publicitária

O período durante a programação em que os comerciais são exibidos. Os intervalos publicitários podem ocorrer antes do conteúdo (pré-lançamento), durante o conteúdo (no meio da publicação) ou depois do conteúdo (pós-lançamento). MediaTailor identifica quebras de anúncios em manifestos por meio de marcadores como. SCTE-35

Servidor de decisão de anúncios (ADS)

Um servidor que fornece especificações de spot publicitário com base em critérios, inclusive campanhas publicitárias e preferências do espectador atuais.

Configuração

Um objeto com MediaTailor o qual você interage. A configuração contém informações de localização sobre o servidor de origem e o servidor de decisão de anúncios (ADS). A configuração também contém endpoints que fornecem pontos de acesso de entrada e saída. MediaTailor

Transcodificação dinâmica

Um processo que compara a qualidade e o formato do anúncio com o conteúdo de vídeo primário quando o conteúdo é solicitado. A transcodificação dinâmica reduz os requisitos de armazenamento e garante que a reprodução faça uma transição perfeita entre o anúncio e o conteúdo do vídeo.

Manipulação manifesta

O processo de reescrever manifestos do servidor de origem, de maneira que os manifestos referenciem os fragmentos de anúncio e conteúdo apropriados. Os anúncios são determinados pela resposta VAST do servidor de decisão de anúncios (ADS). Conforme a reprodução progride, MediaTailor realiza a inserção ou substituição do anúncio no fluxo de conteúdo.

VAST e VMAP

O modelo de veiculação de anúncios em vídeo (VAST) e a lista de reprodução de vários anúncios em vídeo (VMAP) são respostas XML que o servidor de decisão de anúncios envia às solicitações de MediaTailor anúncios. As respostas determinam quais anúncios são MediaTailor inseridos no manifesto. O VMAP também inclui o tempo para intervalos publicitários e os anúncios são disponibilizados dentro deles. Para obter mais informações sobre a lógica por trás da inserção de anúncios do MediaTailor, consulte [Compreender o AWS Elemental MediaTailor comportamento de inserção de anúncios](#). Para obter mais informações sobre como o MediaTailor funciona com VAST, consulte [the section called “Requisitos de integração do servidor de anúncios”](#).

Conceitos de montagem de canais

Aqui está uma visão geral dos conceitos relacionados à montagem de canais.

Canais

Um canal reúne seus manifestos de origem em um fluxo linear. Cada canal tem uma ou mais saídas que contêm URLs de reprodução acessados pelos jogadores. As saídas do canal correspondem às configurações do pacote que você cria para suas fontes de VOD. Um canal contém uma programação, que determina quando as fontes de VOD serão reproduzidas na transmissão do canal.

Configuração do pacote

A configuração do empacotador é uma representação da sua fonte de VOD que contém características específicas do formato do pacote. Você associa as configurações do pacote às saídas do canal para criar fluxos de reprodução para os formatos empacotados da sua fonte de VOD, como HTTP Live Streaming (HLS).

Cronograma

Cada canal é composto por programas organizados na programação do canal. A programação determina o horário em que os programas serão reproduzidos no fluxo linear do canal.

Locais de origem

Um local de origem representa o servidor de origem em que seus ativos estão armazenados. Pode ser o Amazon S3, um servidor HTTP, uma rede de entrega de conteúdo (CDN) ou uma infraestrutura de empacotamento, como MediaPackage.

Fontes de VOD

Uma fonte de VOD representa um único conteúdo, como um filme ou episódio de um programa de TV. Você associa fontes de VOD a programas para adicioná-las ao fluxo linear do seu canal.

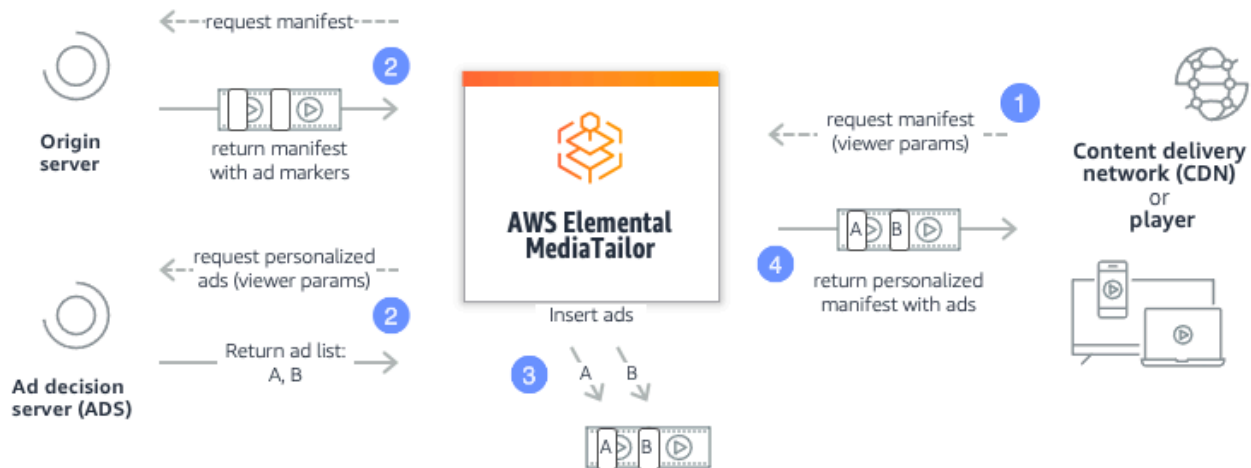
Audiência

Um público define uma coorte de espectadores que, opcionalmente, pode ter conteúdo alternativo. Você pode definir públicos em canais lineares padrão.

Como funciona a inserção de MediaTailor anúncios

AWS Elemental MediaTailor interage entre sua rede de entrega de conteúdo (CDN), servidor de origem e servidor de decisão de anúncios (ADS) para transformar anúncios personalizados em intervalos publicitários em conteúdo ao vivo e em vídeo sob demanda.

Aqui está uma visão geral de como a inserção de MediaTailor anúncios funciona:



1. Um player ou CDN, como a Amazon, CloudFront envia uma solicitação MediaTailor para conteúdo HLS ou DASH. A solicitação contém parâmetros do player com informações sobre o espectador, que são usados para personalização de anúncios.
2. MediaTailor envia uma solicitação ao ADS que contém as informações do visualizador. O ADS escolhe anúncios com base nas informações do visualizador e nas campanhas publicitárias atuais. Ele retorna os URLs para os criativos do anúncio em uma resposta VAST ou VMAP a. MediaTailor

Se você pré-condicionou os anúncios, os URLs apontam para os anúncios pré-transcodificados. Para obter informações sobre a combinação de anúncios com anúncios pré-transcodificados, consulte. [Anúncios pré-condicionados](#)

3. MediaTailor manipula o manifesto para incluir os URLs de anúncios retornados do ADS, transcodificados para corresponder às características de codificação do conteúdo de origem. Se você estiver usando anúncios pré-condicionados, é sua responsabilidade garantir que o anúncio corresponda ao modelo de manifesto.

Se um anúncio ainda não tiver sido transcodificado para corresponder ao conteúdo, ele MediaTailor será ignorado e usado MediaConvert para preparar o anúncio para que esteja pronto para a próxima solicitação.

4. MediaTailor retorna o manifesto totalmente personalizado para o CDN ou player solicitante.

O ADS rastreia os anúncios visualizados com base em marcos de visualização, como início do anúncio, meio do anúncio e fim do anúncio. Conforme a reprodução avança, o player MediaTailor envia ou envia beacons de rastreamento de anúncios para a URL de rastreamento de anúncios do ADS, para registrar quanto de um anúncio foi visualizado. Na inicialização da sessão com MediaTailor, o jogador indica se deve ou MediaTailor deve enviar esses beacons para a sessão.

Para obter informações sobre como começar a inserir anúncios, consulte [Começando com MediaTailor](#).

AWS Elemental MediaTailor fluxo de eventos de inserção de anúncios

AWS Elemental MediaTailor processa a personalização de manifestos para inserção de anúncios no servidor por meio de uma sequência previsível de eventos. A compreensão desse fluxo de eventos ajuda você a compreender como MediaTailor transforma oportunidades de anúncios em experiências de visualização personalizadas e fornece uma base para a solução de problemas quando ocorrem problemas.

Cada oportunidade de inserção de anúncio segue uma sequência cronológica de eventos que MediaTailor registra para visibilidade e monitoramento. Esses eventos representam marcos importantes no processo de personalização do manifesto, desde a detecção de uma oportunidade de anúncio até a entrega de informações de rastreamento.

Sequência de eventos padrão

Quando processa MediaTailor com sucesso uma oportunidade de inserção de anúncio, normalmente ocorre a seguinte sequência de eventos:

1. Detecção de oportunidades de anúncio — MediaTailor detecta um marcador de anúncio (como SCTE-35) no manifesto do conteúdo, indicando que uma oportunidade de personalização de anúncios está disponível.
2. Solicitação do servidor de decisão de anúncio - MediaTailor envia uma solicitação ao servidor de decisão de anúncio (ADS) configurado para recuperar anúncios da oportunidade detectada, incluindo parâmetros do visualizador e duração do intervalo do anúncio.
3. Processamento de respostas de anúncios: MediaTailor recebe uma resposta do ADS contendo informações criativas do anúncio, URLs de rastreamento e metadados sobre os anúncios a serem incluídos. Para obter mais informações sobre os formatos de resposta VAST, VMAP e VPAID, consulte [Requisitos de integração do servidor de anúncios](#).
4. Personalização do manifesto: processa MediaTailor com sucesso a resposta do anúncio, transcodifica os anúncios, se necessário, e gera um manifesto personalizado que inclui URLs

relativos tanto para os segmentos do anúncio quanto para o conteúdo de origem. MediaTailor em seguida, envia o manifesto personalizado para o dispositivo de reprodução.

5. Ativação do beacon de rastreamento - Um beacon de rastreamento é acionado para relatar eventos do anúncio (como impressões, quartis e conclusão) de volta ao servidor de anúncios ou a outros sistemas de medição. No modo de relatório do lado do servidor (padrão), MediaTailor dispara o farol com base nas solicitações do segmento de jogadores. No modo de relatório do lado do cliente, o dispositivo de reprodução dispara o beacon usando URLs fornecidos no manifesto personalizado.

Tempo típico: essa sequência inteira geralmente é concluída em 2 a 5 segundos, dependendo do tempo de resposta do servidor de decisão do anúncio e dos requisitos de processamento criativo do anúncio.

Variações comuns de eventos

Nem todas as tentativas de inserção de anúncios seguem a sequência padrão de sucesso. MediaTailor lida com vários cenários que podem alterar o fluxo de eventos:

Cenário de resposta de anúncio vazio

Quando o servidor de decisão de anúncios não retorna anúncios:

1. A detecção de oportunidades de anúncios ocorre normalmente
2. A solicitação do servidor de decisão de anúncio foi enviada com sucesso
3. O ADS retorna uma resposta vazia com zero anúncios
4. Nenhum anúncio está incluído no manifesto personalizado desta oportunidade

Esse é um cenário normal que pode ocorrer devido à disponibilidade do inventário de anúncios, aos critérios de segmentação ou às regras de negócios configuradas no servidor de decisão de anúncios. Nesse caso, o conteúdo subjacente será exibido, a menos que a transmissão seja uma transmissão ao vivo ou ao vivo para VOD e o slate tenha sido configurado. Para obter mais informações, consulte [MediaTailor arduvia e inserção](#).

Cenários de erro

Quando problemas técnicos impedem a inserção bem-sucedida do anúncio, o fluxo do evento pode ser interrompido em vários pontos:

- Tempo limite do servidor de decisão do anúncio - A solicitação ao ADS excede o limite de tempo limite configurado, impedindo a recuperação do anúncio.
- Erros de comunicação - problemas de rede ou conectividade MediaTailor impedem o acesso ao servidor de decisão do anúncio.
- Erros de análise de resposta - O ADS retorna uma resposta que MediaTailor não pode ser analisada devido ao formato ou estrutura VAST inválidos.

Em cenários de erro, MediaTailor normalmente continua com a reprodução do conteúdo sem anúncios, garantindo uma experiência ininterrupta para o espectador. Para transmissões ao vivo ou ao vivo para VOD com o slate configurado, o conteúdo do slate pode ser exibido em vez disso. Para obter mais informações, consulte [MediaTailor ardósia e inserção](#).

Cenários de redirecionamento VAST

Quando a resposta inicial do VAST contém um redirecionamento para outro servidor de anúncios:

1. A detecção de oportunidades de anúncio e a solicitação inicial de ADS ocorrem normalmente
2. A resposta inicial contém uma instrução de redirecionamento
3. MediaTailor segue o redirecionamento para recuperar a resposta final do anúncio
4. A personalização do manifesto prossegue com o conteúdo final do anúncio
5. A ativação do farol de rastreamento ocorre normalmente

Os redirecionamentos VAST são comuns na publicidade programática e permitem o encadeamento de servidores de anúncios e a tomada de decisões em tempo real. MediaTailor permite até 7 redirecionamentos de wrapper e redirecionamentos HTTP ilimitados durante o processo de recuperação de anúncios.

Visão geral do cronograma do evento

Compreender as características temporais dos fluxos de MediaTailor eventos ajuda a definir expectativas adequadas para o desempenho da inserção de anúncios:

- Detecção de oportunidades de anúncio - ocorre quando MediaTailor recebe uma `GetManifest` solicitação do player e encontra os marcadores de anúncio no manifesto do conteúdo.
- Interação com o servidor de decisão de anúncios: normalmente leva de 100 a 500 milissegundos, dependendo do tempo de resposta do ADS e das condições da rede.

- Personalização do manifesto - geralmente é concluída dentro de 50 a 200 milissegundos após o recebimento da resposta do ADS.
- Tempo do farol de rastreamento - Varia de acordo com o modo de relatório. Para beacons do lado do servidor, o tempo é baseado nas solicitações de segmento do player cliente; os beacons do lado do cliente são acionados com base na implementação do player.

Considerações de desempenho: o tempo total desde a detecção da oportunidade de anúncio até a entrega do manifesto normalmente deve permanecer abaixo de 5 segundos para manter a experiência ideal do espectador. Os componentes individuais são concluídos em milissegundos, conforme descrito acima. No entanto, o limite de 5 segundos é responsável por possíveis valores de tempo limite, tentativas de repetição e variabilidade da rede que podem ocorrer durante todo o fluxo de trabalho de inserção de anúncios. Atrasos maiores podem indicar problemas de desempenho do ADS ou problemas de conectividade de rede.

Usando o conhecimento do fluxo de eventos

A compreensão MediaTailor do fluxo de eventos fornece a base para:

- Planejamento da implementação - Conhecer a sequência do evento ajuda a projetar a integração do player e a configuração do servidor de anúncios.
- Otimização de desempenho - Compreender as expectativas de tempo permite a identificação de gargalos e oportunidades de otimização.
- Preparação para solução de problemas - A familiaridade com os fluxos normais de eventos facilita a identificação de quando algo dá errado.

Para obter orientações detalhadas sobre solução de problemas usando a análise de fluxo de eventos, consulte [Solução de problemas](#). Para obter detalhes técnicos sobre registro e monitoramento de eventos, consulte [Visualizar logs](#).

Serviços relacionados

- CloudFrontA Amazon é um serviço global de rede de entrega de conteúdo (CDN) que entrega dados e vídeos com segurança aos seus espectadores. Use o CloudFront para entregar conteúdo com o melhor desempenho possível. Para obter mais informações sobre CloudFront, consulte o [CloudFrontsite da Amazon](#).

- AWS Elemental MediaPackage é um serviço de empacotamento e originação just-in-time que personaliza ativos de vídeo ao vivo para distribuição em um formato compatível com o dispositivo que faz a solicitação. Use o AWS Elemental MediaPackage como um servidor de origem para preparar conteúdo e adicionar marcadores de anúncios antes de enviar streamings para o MediaTailor. Para obter mais informações sobre como MediaTailor funciona com servidores de origem, consulte [Como funciona a inserção de MediaTailor anúncios](#).
- AWS Identity and Access Management (IAM) é um serviço web que ajuda você a controlar com segurança o acesso aos AWS recursos para seus usuários. Use o IAM para controlar quem pode usar seus AWS recursos (autenticação) e quais recursos eles podem usar de quais maneiras (autorização). Para obter mais informações, consulte [Configurar AWS Elemental MediaTailor](#).

Acessando MediaTailor

Você pode acessar MediaTailor usando o console do serviço.

Acesse seu Conta da AWS fornecendo credenciais que verificam se você tem permissões para usar os serviços.

Para fazer login no MediaTailor console, use o seguinte link: **<https://console.aws.amazon.com/mediatailor/home>**.

Preços para MediaTailor

Assim como em outros AWS produtos, não há contratos ou compromissos mínimos de uso MediaTailor. Você é cobrado com base no uso do serviço. Para obter mais informações, consulte [Preços do MediaTailor](#).

Regiões para MediaTailor

Para reduzir a latência de dados em seus aplicativos, MediaTailor oferece endpoints regionais para fazer suas solicitações. Para ver a lista de regiões nas quais MediaTailor está disponível, consulte [Endpoints regionais](#).

Configurar AWS Elemental MediaTailor

Esta seção orienta você sobre as etapas necessárias para configurar os usuários para acessar o AWS Elemental MediaTailor. Para obter informações básicas e adicionais sobre gerenciamento de identidade e acesso para MediaTailor, consulte [Identity and Access Management para AWS Elemental MediaTailor](#).

Para começar a usar AWS Elemental MediaTailor, conclua as etapas a seguir.

Tópicos

- [Inscreva-se para um Conta da AWS](#)

Inscreva-se para um Conta da AWS

Para começar AWS, você precisa de um Conta da AWS. Para obter informações sobre como criar um Conta da AWS, consulte [Introdução a um Conta da AWS](#) no Guia de AWS Gerenciamento de contas referência.

Começando com MediaTailor

Para começar MediaTailor, você pode escolher entre dois tutoriais: um sobre como configurar a inserção de anúncios e outro sobre montagem de canais. Você também pode explorar um workshop prático para eventos ao vivo. O tutorial de inserção de anúncios guiará você no envio de uma solicitação de reprodução para MediaTailor incorporar anúncios personalizados ao seu fluxo de conteúdo. O tutorial de montagem de canais mostrará como visualizar a transmissão do seu canal, incluindo os anúncios personalizados, diretamente em um navegador usando um URL de reprodução.

Você também pode explorar o [Workshop do Live Event Framework](#). Este workshop prático orienta você por todo o ciclo de vida de uma configuração de evento AWS Elemental MediaTailor ao vivo, desde a descoberta e implantação iniciais até a otimização e o escalonamento. Você pode acompanhar o workshop para obter uma configuração simples que usa uma MediaTailor configuração para todos os eventos e canais, ou para uma arquitetura avançada que usa configurações dedicadas por evento ou canal.

Tópicos

- [Introdução à inserção MediaTailor de anúncios](#)
- [Introdução à montagem MediaTailor de canais](#)

Introdução à inserção MediaTailor de anúncios

Para usar AWS Elemental MediaTailor, você precisa de permissões Conta da AWS e para acessar, visualizar e editar MediaTailor configurações. Para obter informações sobre como fazer isso, consulte [Configurar AWS Elemental MediaTailor](#).

Este tutorial de conceitos básicos mostra como executar as seguintes tarefas:

- Prepare seus fluxos de conteúdo HLS ou DASH
- Configurar um URL de modelo de servidor de decisão de anúncios (ADS)
- Crie uma MediaTailor configuração que contenha um endpoint de reprodução
- Use seu player ou rede de distribuição de conteúdo (CDN) para fazer uma solicitação de reprodução para MediaTailor

Para uma configuração avançada de URL do ADS usando variáveis dinâmicas e parâmetros do player, consulte [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#). Para obter informações sobre a transmissão de parâmetros por meio de CDNs, consulte [MediaTailor parâmetros de consulta do manifesto](#).

Quando terminar, você poderá enviar uma solicitação de reprodução MediaTailor para conteúdo de anúncio personalizado em seu stream.

Tópicos

- [Pré-requisitos](#)
- [Etapa 1: Acesso AWS Elemental MediaTailor](#)
- [Etapa 2: preparar um stream](#)
- [Etapa 3: Configurar o URL de solicitação do ADS e os parâmetros de consulta](#)
- [Etapa 4: Criar uma configuração](#)
- [Etapa 5: testar a configuração](#)
- [Etapa 6: enviar a solicitação de reprodução para AWS Elemental MediaTailor](#)
- [Etapa 7 \(opcional\): Monitor AWS Elemental MediaTailor atividade](#)
- [Etapa 8: Limpeza](#)

Pré-requisitos

Antes de começar, é necessário concluir as etapas em [Configurar AWS Elemental MediaTailor](#).

Etapa 1: Acesso AWS Elemental MediaTailor

Usando suas credenciais do IAM, faça login no MediaTailor console em **`https://console.aws.amazon.com/mediatailor/home`**.

Etapa 2: preparar um stream

Configure o servidor de origem a fim de produzir manifestos para HLS ou DASH compatíveis com o AWS Elemental MediaTailor. Seu servidor de origem deve usar portas HTTP padrão (porta 80) ou portas HTTPS (porta 443). MediaTailor não oferece suporte a portas personalizadas para comunicação com o servidor de origem.

Prepare um fluxo HLS

Os manifestos HLS devem atender aos seguintes requisitos:

- Os manifestos devem ser acessíveis na internet pública.
- Os manifestos devem ser ao vivo ou vídeo sob demanda (VOD).
- Os manifestos devem ter um EXT-X-VERSION de 3 ou superior.
- Para conteúdo ao vivo, os manifestos devem conter marcadores para delinear disponibilidades de anúncios. Isso é opcional para conteúdo VOD, que pode usar deslocamentos de tempo VMAP em vez disso.

O arquivo manifesto deve ter slots de anúncios marcados com um dos seguintes:

- #EXT-X-CUE-OUT /# EXT-X-CUE-IN (mais comum) com durações, conforme mostrado no exemplo a seguir.

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

- #EXT-X-DATERANGE (menos comum) com durações, conforme mostrado no exemplo a seguir.

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

Todos os campos mostrados para #EXT-X-DATERANGE são obrigatórios.

A maneira como você configura os marcadores de anúncios no manifesto influencia se os anúncios são inseridos em um streaming ou substituem outros fragmentos no streaming. Para obter mais informações, consulte [the section called “Compreender o comportamento de inserção de anúncios”](#).

- As playlists multivariantes do HLS devem seguir a especificação HLS documentada em [HTTP live streaming: Master playlist tags](#). Em especial, #EXT-X-STREAM-INF deve incluir os campos RESOLUTION, BANDWIDTH e CODEC.

Depois de configurar o stream, observe o prefixo do URL de origem do conteúdo para a playlist multivariante. Você precisa dele para criar a configuração no AWS Elemental MediaTailor, posteriormente neste tutorial.

Prepare um stream DASH

Os manifestos DASH devem atender aos seguintes requisitos:

- Os manifestos devem ser acessíveis na internet pública.
- Os manifestos devem ser ao vivo ou vídeo sob demanda (VOD).
- Os manifestos devem marcar eventos como disponibilidades de anúncios usando marcadores de inserção por emenda ou marcadores de sinal de tempo. Você pode fornecer os marcadores de anúncios em XML limpo ou em binário codificado em base64. Para inserção por emenda, o indicador fora de rede deve ser habilitado. Para marcadores de sinal de tempo, o ID do tipo de segmentação, localizado dentro do UPID de segmentação, deve ser um valor de saída reconhecido por AWS Elemental MediaTailor. A disponibilidade de anúncios começa no início do evento e dura pela duração do evento, se especificada, ou até que o próximo evento comece.

O exemplo a seguir mostra um evento designado como uma disponibilidade de anúncios usando marcadores de inserção por emenda. A duração dessa disponibilidade de anúncios é a duração do evento.

```
<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>
```

- Disponibilidades de anúncios devem ter as mesmas configurações `AdaptationSet` e `Representation` de streamings de conteúdo. O AWS Elemental MediaTailor usa essas configurações para transcodificar os anúncios de acordo com o streaming do conteúdo, tendo em vista uma alternância suave entre os dois.

Depois que você configurar o streaming, anote o prefixo do URL de origem do conteúdo do manifesto DASH. Você precisa dela para criar a configuração AWS Elemental MediaTailor, posteriormente neste tutorial.

Etapa 3: Configurar o URL de solicitação do ADS e os parâmetros de consulta

Para determinar os parâmetros de consulta exigidos pelo ADS, gere um URL da tag de anúncio com base no ADS. Esse URL funciona como um modelo para solicitações feitas ao ADS e consiste no seguinte:

- Valores estáticos
- Valores gerados por AWS Elemental MediaTailor (indicados por `session` ou parâmetros de `avail` consulta)
- Valores gerados por players, obtidos do aplicativo cliente (indicados por parâmetros de consulta `player_params`.)

Example URL de tag de anúncio de um ADS

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

Em que:

- `output` e `content_id` são valores estáticos
- `playerSession= [session.id]` é um valor dinâmico fornecido por. AWS Elemental MediaTailor O valor de `[session.id]` muda para cada sessão de player e resulta em um URL diferente para a solicitação VAST de cada sessão.
- `cust_params` são valores dinâmicos fornecidos pelo player

A solicitação de playlist multivariada do player deve fornecer pares de valores-chave que correspondam aos parâmetros de `player_params`. consulte no URL da solicitação do ADS. Para obter mais informações sobre como configurar pares de valores-chave na solicitação para AWS Elemental MediaTailor, consulte. [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#)

Insira o URL “modelo” configurado ao criar o server/ADS mapeamento de origem em MediaTailor, em [Etapa 4: Criar uma configuração](#).

Teste

Use uma resposta VAST estática do ADS para fins de teste. O ideal é que a resposta VAST retorne uma versão MP4 de qualidade de mezanino que o AWS Elemental MediaTailor pode transcodificar. Se a resposta do ADS contiver várias representações de reprodução, MediaTailor selecionará a representação MP4 da mais alta qualidade e resolução e a enviará para o transcodificador.

Etapa 4: Criar uma configuração

A AWS Elemental MediaTailor configuração contém informações de mapeamento para o servidor de origem e o ADS.

Para criar uma configuração (console)

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. Na página Configurações, selecione Criar configuração.
3. Na seção Configuration (Configuração) na parte inferior da página, em Configuration name (Nome da configuração), digite um nome exclusivo que descreva a configuração. O nome é o principal identificador da configuração. O tamanho máximo permitido é de 512 caracteres.
4. Em Fonte de conteúdo de vídeo, insira o prefixo do URL da playlist multivariante HLS ou do manifesto DASH desse stream, menos o ID do ativo. Por exemplo, se o URL da playlist multivariante for `http://origin-server.com/a/master.m3u8`, você digitaria `http://origin-server.com/a/`. Também é possível inserir um prefixo mais curto, como `http://origin-server.com`, mas você deve incluir o `/a/` no ID do ativo na solicitação do player para o conteúdo. O tamanho máximo é de 512 caracteres.

 Note

Caso a origem do conteúdo use HTTPS, o certificado deve ser de uma autoridade de certificação bem conhecida. (Ele não pode ser um certificado autoassinado.) Caso contrário, AWS Elemental MediaTailor não conseguirá se conectar à origem do conteúdo e não poderá veicular manifestos em resposta às solicitações dos jogadores.

5. Em Servidor de decisão de anúncios, insira o URL do seu ADS. Esse é o URL com variáveis conforme descrito em [Etapa 3: Configurar o URL de solicitação do ADS e os parâmetros de consulta](#) ou o URL VAST estático que você está usando para fins de teste. O tamanho máximo é de 25.000 caracteres.

 Note

Caso o ADS use HTTPS, o certificado deve ser de uma autoridade de certificação bem conhecida. (Ele não pode ser um certificado autoassinado.) O mesmo acontece com URLs de anúncios de mezanino retornados pelo ADS. Do contrário, o MediaTailor deixa de recuperar e unir anúncios aos manifestos na origem do conteúdo.

6. (Opcional conforme necessário para o DASH) Em Localização, escolha DISABLED se você tiver regras de roteamento de CDN configuradas para acessar MediaTailor manifestos e estiver usando relatórios do lado do cliente ou se seus jogadores aceitarem redirecionamentos HTTP fixos.

Para obter mais informações sobre o recurso Location (Local), consulte [the section called “Recurso de localização”](#).

7. (Opcional) Se o seu servidor de origem produz um manifesto DASH de período único, escolha DASH mpd manifest origin type (Tipo de origem do manifesto mpd DASH) e escolha SINGLE_PERIOD. Por padrão, MediaTailor trata os manifestos do DASH como manifestos de vários períodos. Para obter mais informações, consulte [the section called “Integrando uma fonte MPEG-DASH”](#).
8. Escolha Criar configuração.

AWS Elemental MediaTailor exibe a nova configuração na página Configurações.

Etapa 5: testar a configuração

Depois de salvar a configuração, teste o streaming usando um URL no formato apropriado ao protocolo de streaming:

- Exemplo: HLS

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

- Exemplo: DASH

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

Em que:

- `playback-endpoint` é o endpoint de reprodução exclusivo gerado pelo AWS Elemental MediaTailor quando a configuração foi criada.

Exemplo

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

- `hashed-account-id` é sua Conta da AWS identidade.

Exemplo

```
777788889999
```

- `origin-id` é o nome dado quando se cria a configuração.

Exemplo

```
myOrigin
```

- `master.m3u8` ou `manifest.mpd` é o nome do manifesto do streaming de teste mais a extensão do arquivo. Defina isso para que você obtenha um manifesto totalmente identificado ao anexá-lo à origem do conteúdo do vídeo configurada em [the section called “Etapa 4: Criar uma configuração”](#).

Usando os valores dos exemplos anteriores, os URLs completos são os seguintes.

- Exemplo: HLS

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

- Exemplo: DASH

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

Teste o streaming usando um dos seguintes métodos.

- Conforme mostrado no exemplo anterior, digite o URL em um player autônomo.
- Teste o streaming no próprio ambiente do player.

Etapa 6: enviar a solicitação de reprodução para AWS Elemental MediaTailor

Configure o player downstream ou a CDN para enviar solicitações de reprodução ao endpoint de reprodução da configuração fornecido pelo AWS Elemental MediaTailor. Todas as variáveis dinâmicas definidas pelo player usadas no URL de solicitação ADS em [Etapa 3: Configurar o URL de solicitação do ADS e os parâmetros de consulta](#) devem ser definidas na solicitação de manifesto do player.

Example

Vamos supor que o URL ADS do modelo seja o seguinte.

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

Em seguida, defina `[player_params.cust_params]` na solicitação do player prefaciando o par de valores-chave com `ads.` AWS Elemental MediaTailor passa parâmetros que não são precedidos `ads.` para o servidor de origem em vez do ADS.

O URL de solicitação do player é uma variação dos seguintes exemplos HLS e DASH,

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

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

Quando AWS Elemental MediaTailor recebe a solicitação do jogador, ele define as variáveis do jogador com base nas informações da solicitação. O URL da solicitação ADS resultante é uma variação disso.

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

Para obter mais informações sobre como configurar pares de chave/valor a serem passados ao ADS, consulte [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#).

Etapa 7 (opcional): Monitor AWS Elemental MediaTailor atividade

Use a Amazon CloudWatch e o Amazon CloudWatch Logs para rastrear AWS Elemental MediaTailor atividades, como contagens de solicitações, erros e anúncios publicados.

Se esta é a primeira vez que você usa CloudWatch com AWS Elemental MediaTailor, crie uma função AWS Identity and Access Management (IAM) para permitir a comunicação entre os serviços.

Para permitir AWS Elemental MediaTailor acesso a CloudWatch (console)

1. Abra o console do IAM em <https://console.aws.amazon.com/iam/>.
2. No painel de navegação do console do IAM, escolha Perfis e, em seguida, Criar perfil.
3. Escolha o outro tipo de Conta da AWS função.
4. Em ID da conta, insira seu Conta da AWS ID.
5. Selecione Require external ID (Exigir ID externo) e digite **midas**. Essa opção adiciona automaticamente uma condição à política de confiança que só permite ao serviço assumir a função caso a solicitação inclua o `sts:ExternalID` correto.
6. Escolha Próximo: Permissões.
7. Adicione uma política de permissões que especifique quais ações essa função pode concluir. Selecione uma das seguintes opções e escolha Next: Review (Avançar: revisar):

- CloudWatchLogsFullAccess para fornecer acesso total ao Amazon CloudWatch Logs
 - CloudWatchFullAccess para fornecer acesso total à Amazon CloudWatch
8. Em Role name (Nome da função), digite **MediaTailorLogger** e escolha Create role (Criar função).
 9. Na página Roles (Funções), selecione a função recém-criada.
 10. Edite a relação de confiança para atualizar a entidade principal:
 1. Na página Summary (Resumo) da função, escolha a guia Trust relationship (Relação de confiança).
 2. Selecione Editar relação de confiança.
 3. No documento da política, altere a entidade principal para o serviço AWS Elemental MediaTailor . A aparência deve ser semelhante à seguinte.

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

Toda a política deve ser lida da seguinte maneira.

4. Selecione Atualizar política de confiança.

Etapa 8: Limpeza

Para evitar custos adicionais, exclua todas as configurações desnecessárias.

Para excluir uma configuração (console)

1. Na página AWS Elemental MediaTailor Configurações, faça o seguinte:
 - Escolha o Configuration name (Nome da configuração) que deseja excluir.
 - Na coluna Configuration name (Nome da configuração), escolha o botão de opção e Delete (Excluir).
2. Na caixa de confirmação Delete configuration (Excluir configuração), digite **Delete** e escolha Delete (Excluir) novamente.

AWS Elemental MediaTailor remove a configuração.

Introdução à montagem MediaTailor de canais

Este tutorial de conceitos básicos mostra como executar as seguintes tarefas:

- Crie um local de origem e adicione conteúdo de origem a ele
- Criar um canal
- Crie uma lista de programas para reproduzir o conteúdo do seu canal de acordo com uma programação
- Adicione anúncios personalizados ao stream do canal usando a inserção de AWS Elemental MediaTailor anúncios

Ao terminar, você poderá abrir um navegador, inserir o URL de reprodução do seu canal e ver a transmissão do seu canal contendo anúncios personalizados.

Este tutorial mostra as etapas básicas para começar a montar MediaTailor canais. Para obter informações mais avançadas, consulte [Usando AWS Elemental MediaTailor para criar fluxos montados lineares](#).

Custo estimado

- A taxa de um canal ativo é de \$0,10 por hora. Você não é cobrado por canais inativos.

Tópicos

- [Pré-requisitos](#)
- [Etapa 1: criar um local de origem](#)
- [Etapa 2: Adicionar fontes de VOD ao seu local de origem](#)
- [Etapa 3: criar um canal](#)
- [Etapa 4: adicionar programas à programação do seu canal](#)
- [Etapa 5 \(optional\): Use MediaTailor para inserir anúncios personalizados em seu stream](#)
- [Etapa 6: inicie seu canal](#)
- [Etapa 7: teste seu canal](#)
- [Etapa 8: Limpeza](#)

Pré-requisitos

Antes de começar este tutorial, você deve preencher estes requisitos:

- Certifique-se de ter concluído as etapas em [Configurar AWS Elemental MediaTailor](#).
- Você deve ter ativos disponíveis tanto para o conteúdo de origem do VOD quanto para a lista de anúncios. Você deve conhecer o caminho até os manifestos dos ativos.

Note

Se você estiver usando taxa de bits adaptativa automática (ABR) ou codificação por título, deverá codificar seus ativos para que todas as variantes tenham o mesmo tamanho e tenham o mesmo número de faixas secundárias. Recomendamos que você use um modelo de codificação com um comprimento mínimo de segmento de um segundo.

Etapa 1: criar um local de origem

Um local de origem representa o servidor de origem em que seu conteúdo está armazenado. Pode ser o Amazon S3, um servidor web padrão, uma rede de entrega de conteúdo (CDN) ou uma origem de embalagem, como AWS Elemental MediaPackage.

MediaTailor busca os manifestos de conteúdo do seu local de origem e os usa para montar uma janela de manifesto deslizante ativa que faz referência aos segmentos de conteúdo subjacentes.

Para criar um local de origem, execute o procedimento a seguir.

Para criar um local de origem

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem do canal > Locais de origem.
3. Na barra de navegação, escolha Criar local de origem.
4. Em Configuração do local de origem, insira um identificador e a localização do conteúdo de origem:
 - Nome: um identificador para seu local de origem, como my-origin.

- URL base: a URL base do servidor de origem em que seu conteúdo está hospedado, como <https://111111111111.cloudfront.net>. O URL deve estar em um formato de URL HTTP padrão, prefixado com <http://> ou <https://>.

5. Escolha Criar local de origem.

Etapa 2: Adicionar fontes de VOD ao seu local de origem

Agora que você definiu um ou mais locais de origem para seu canal, você pode adicionar uma ou mais fontes de VOD. Cada fonte de VOD representa um único conteúdo, como um único filme, um episódio de um programa de TV ou um clipe de destaque.

Você deve criar pelo menos uma configuração de pacote para sua fonte de VOD. Cada configuração de pacote contém o formato do pacote e as configurações de manifesto para suas fontes de VOD. Em seguida, você adiciona as configurações do pacote ao seu canal para criar saídas.

Você pode usar várias configurações de pacotes para criar diferentes saídas de canal. Por exemplo, se sua fonte de VOD estiver empacotada como HLS e DASH, você poderá criar duas configurações de pacote para cada formato. Em seguida, você pode usar os grupos de origem da configuração do pacote para criar duas saídas de canal: uma para HLS e outra para DASH.

Para adicionar fontes de VOD e criar configurações de pacotes

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem do canal > Locais de origem.
3. No painel Locais de origem, escolha o local de origem que você criou no [Para criar um local de origem](#) procedimento.
4. Escolha Adicionar fonte VOD.
5. Em Detalhes da fonte de VOD, insira um nome para sua fonte de VOD, como my-example-video.
6. Em Package configurations > *source-group-name* insira as informações sobre a configuração do pacote:

Note

Todas as configurações do pacote de origem devem ter a mesma duração, conforme determinado pelo manifesto da fonte. Além disso, todas as fontes em uma configuração de pacote devem ter o mesmo número de fluxos secundários. Para atender a esses

requisitos, recomendamos que você use um modelo de codificação para seus ativos. Recomendamos que você use um modelo de codificação com um comprimento mínimo de segmento de um segundo. MediaTailor não oferece suporte por título ou streaming automático de taxa de bits adaptativa (ABR) porque esses métodos de codificação violam esses requisitos.

- Grupo de origem: insira um nome de grupo de origem que descreva essa configuração de pacote, como HLS-4k. Anote esse nome; você o referenciará ao criar a saída do seu canal. Para obter mais informações, consulte [Usando grupos de origem com as saídas do seu canal](#).
 - Tipo: Selecione o formato do pacote para essa configuração. MediaTailor suporta HLS e DASH.
 - Caminho relativo: o caminho relativo do URL HTTP base do local de origem até o manifesto. Por exemplo, /my/path/index.m3u8.
7. Escolha Add source (Adicionar origem).
 8. Repita as etapas de 4 a 7 neste procedimento para adicionar a fonte de VOD à sua lista de anúncios.

Etapa 3: criar um canal

Um canal reúne suas fontes em uma transmissão linear ao vivo. Cada canal contém uma ou mais saídas que correspondem às configurações do pacote da sua fonte VOD.

Primeiro, você cria um canal e depois adiciona suas fontes de VOD à programação do canal criando programas.

Para criar um canal

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. Na barra de navegação, escolha Criar canal.
4. Em Detalhes do canal, insira detalhes sobre seu canal:
 - Nome: insira um nome para seu canal.
 - Modo de reprodução: determina quais tipos de transições de programa são permitidas e o que acontece com um programa após sua conclusão. Use o modo de loop padrão.

5. Escolha Próximo.
6. Em Detalhes da saída, defina as configurações dessa saída:
 - Nome do manifesto: insira um nome de manifesto, como ***index***. MediaTailor anexará a extensão do formato, como .m3u8 para HLS.

 Note

Você deve inserir um nome de manifesto exclusivo por saída do canal.

- Tipo de formato: selecione o formato de streaming para o canal. DASH e HLS são suportados. Escolha o formato que corresponde à configuração do pacote que você criou [Etapa 1: criar um local de origem](#).
 - Grupo de origem: insira o nome do grupo de origem que você criou em [Etapa 1: criar um local de origem](#).
7. Em Configurações do manifesto, insira informações adicionais sobre suas configurações de manifesto:
 - Janela de manifesto (segundos): a janela de tempo (em segundos) contida em cada manifesto. O valor mínimo é 30 segundos e o valor máximo é 3600 segundos.
 8. Escolha Próximo.
 9. Em Política do canal, selecione Não anexar política do canal. Essa opção restringe a reprodução somente para aqueles que têm acesso às suas Conta da AWS credenciais.
 10. Escolha Próximo.
 11. Revise suas configurações no painel Revisar e criar.
 12. Escolha Create channel (Criar canal).

 Note

Os canais são criados em um estado parado. Seu canal não estará ativo até que você o inicie.

Etapa 4: adicionar programas à programação do seu canal

Agora que você tem um canal, você adicionará programas à programação do canal. Cada programa contém uma fonte de VOD de um local de origem em sua conta. A programação do canal determina a ordem em que seus programas serão reproduzidos na transmissão do canal.

Cada programa pode ter um ou mais intervalos publicitários. Você insere um intervalo de anúncio especificando uma fonte de VOD para usar como lista de anúncios. A duração do intervalo publicitário é determinada pela duração da lista. Opcionalmente, você pode usar um servidor de inserção de anúncios do lado do servidor, como inserção de MediaTailor anúncios, para personalizar seus intervalos publicitários.

Para adicionar programas à programação do seu canal

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. No painel Canais, escolha o canal que você criou no [Etapa 3: criar um canal](#) procedimento.
4. Em Detalhes do programa, insira detalhes sobre seu programa:
 - Nome: Esse é o nome do programa a ser adicionado à programação do seu canal.
 - Nome do local de origem: escolha Selecionar um local de origem existente e selecione o local de origem que você criou no [Etapa 1: criar um local de origem](#) menu suspenso Selecionar um local de origem.
 - Nome da fonte de VOD: escolha Selecionar uma fonte de VOD existente e selecione a fonte de VOD que você criou anteriormente neste tutorial.
5. Em Configuração de reprodução, defina como e quando um programa é inserido na programação de um canal:
 - Tipo de transição: esse valor é fixado em Relativo. O tipo de transição relativa indica que esse programa ocorre em relação a outros programas na lista de programas.
 - Posição relativa: se esse for o primeiro programa na programação do seu canal, você pode pular essa configuração. Se não for o primeiro programa na programação do seu canal, escolha onde anexar o programa na lista de programas. Você pode selecionar Antes do programa ou Depois do programa.
 - Programa relativo: se esse for o primeiro programa em sua agenda, você pode pular essa configuração. Se não for o primeiro programa na programação do seu canal, escolha Usar

programa existente e selecione o nome do programa que você criou em [Para adicionar programas à programação do seu canal](#).

6. Selecione Adicionar intervalo publicitário. Em Intervalos publicitários, defina as configurações do intervalo publicitário:

- Nome do local de origem do Slate: Escolha Selecionar um local de origem existente e escolha o local de origem em que seu Slate está armazenado, criado anteriormente neste tutorial.
- Nome da fonte de VOD: escolha Selecionar uma fonte de VOD existente e escolha a fonte de VOD que você está usando para o slate que você adicionou anteriormente neste tutorial. A duração da lista determina a duração do intervalo publicitário.
- Para deslocamento em milissegundos: esse valor determina o horário de início do intervalo publicitário em milissegundos, como um deslocamento em relação ao início do programa. Insira qualquer valor menor que a duração da fonte de VOD e que esteja alinhado com um limite de segmento em todas as faixas da fonte VOD do programa (todas as faixas de áudio, vídeo e legendas ocultas), caso contrário, o intervalo do anúncio será ignorado. Por exemplo, se você inserir 0, isso criará um intervalo publicitário pré-lançamento que será reproduzido antes do início do programa. Nota:.

7. Escolha Adicionar programa.

Para obter mais informações sobre programas, consulte [Configuring ad breaks for your program](#).

Para obter informações mais avançadas sobre o uso de anúncios com seu fluxo linear, consulte [Configurações opcionais](#).

Etapa 5 (optional): Use MediaTailor para inserir anúncios personalizados em seu stream

Agora você tem um canal com programas. Se quiser, você pode usar MediaTailor para inserir anúncios personalizados nos intervalos publicitários de seus programas no stream do canal.

Pré-requisitos

Antes de continuar, você deve atender aos seguintes requisitos:

- Você deve ter um servidor de decisão de anúncios (ADS).
- Você deve ter definido as configurações de Ad Break no [Adicionar um programa à programação de um canal](#) procedimento.

Para adicionar anúncios personalizados ao stream do seu canal usando MediaTailor

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Configurações.
3. Em Configurações obrigatórias, insira as informações básicas necessárias sobre sua configuração:
 - Nome: O nome da sua configuração.
 - Fonte de conteúdo: insira o URL de reprodução da saída do seu canal, menos o nome do arquivo e a extensão. Para obter informações avançadas sobre MediaTailor configuração, consulte [Configurações necessárias](#).
 - Adicionar servidor de decisão: insira o URL do seu ADS.
4. Opcionalmente, você pode definir os aliases de configuração, os detalhes de personalização e as configurações avançadas. Para obter informações sobre essas configurações, consulte [Configurações opcionais](#).
5. Na barra de navegação, escolha Criar configuração.

Para obter informações mais avançadas sobre o uso da inserção de MediaTailor anúncios, consulte [Utilizar AWS Elemental MediaTailor para inserir anúncios](#).

Etapa 6: inicie seu canal

Agora você tem um canal. Mas antes de acessar a transmissão do canal, você precisa iniciá-lo. Se você tentar acessar um canal antes que ele esteja ativo, MediaTailor retornará um código de 4xx erro HTTP.

Comece seu canal

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. Na barra de navegação, escolha Iniciar.

Etapa 7: teste seu canal

Para verificar se seu canal está funcionando corretamente, abra um navegador da Web e insira o URL da saída do seu canal. Você deve ver o stream do seu canal.

Em alguns casos, talvez você precise limpar o cache para ver o comportamento esperado.

Etapa 8: Limpeza

Depois de terminar com o canal que você criou para este tutorial, você deve limpá-lo excluindo o canal.

Você deixará de incorrer em cobranças por esse canal assim que o status do canal mudar para Interrompido. Para manter seu canal para mais tarde, mas não incorrer em cobranças, você pode interromper o canal agora e reiniciá-lo mais tarde.

Para excluir seu canal

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. Selecione o canal que você deseja excluir.
4. Se seu canal estiver em execução, no menu suspenso Ações, escolha Parar. Você deve interromper seu canal antes de excluí-lo.
5. Quando seu canal estiver parado, no menu suspenso Ações, escolha Excluir.

Utilizar AWS Elemental MediaTailor para inserir anúncios

Configuração é um objeto com que você interage no AWS Elemental MediaTailor. A configuração contém as informações de mapeamento do servidor de origem e do servidor de decisão de anúncios (ADS). Você também pode definir uma reprodução padrão MediaTailor para ser usada quando um anúncio não está disponível ou não preenche toda a disponibilidade do anúncio.

Se você usa uma rede de distribuição de conteúdo (CDN) com MediaTailor, deve configurar as regras de comportamento na CDN antes de adicionar informações da CDN à configuração. Para obter mais informações sobre como configurar a CDN, consulte [Usando uma CDN para otimizar a personalização de MediaTailor anúncios e a entrega de conteúdo](#).

Tópicos

- [Codecs de áudio e vídeo compatíveis](#)
- [Compreender o AWS Elemental MediaTailor comportamento de inserção de anúncios](#)
- [MediaTailor visão geral e implementação da inserção de anúncios guiada pelo servidor](#)
- [MediaTailor requisitos de integração do servidor de anúncios](#)
- [MediaTailor gerenciamento de configuração de reprodução](#)
- [Integrando uma fonte de conteúdo para inserção de MediaTailor anúncios](#)
- [Integrar AWS Elemental MediaTailor com as plataformas de anúncios do Google](#)
- [Personalizando o comportamento de anúncios com a supressão de anúncios](#)
- [MediaTailor pára-choques e inserção](#)
- [MediaTailor pré-rolô e inserção](#)
- [MediaTailor ardósia e inserção](#)
- [Transcodificação de anúncios em AWS Elemental MediaTailor](#)
- [Pré-busca de anúncios](#)
- [Usando anúncios pré-condicionados com AWS Elemental MediaTailor](#)
- [Trabalhando com funções de MediaTailor monetização](#)
- [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#)
- [MediaTailor parâmetros de consulta do manifesto](#)
- [Dados de relatórios e rastreamento](#)
- [Anúncios de sobreposição](#)
- [Decoração de ID do anúncio](#)

Codecs de áudio e vídeo compatíveis

MediaTailor suporta os seguintes codecs.

- Codecs de áudio: mp4a,, e ac-3 ec-3
- Codecs de vídeo: h.264 (AVC), h.265 (HEVC), av01 () AV1

Compreender o AWS Elemental MediaTailor comportamento de inserção de anúncios

AWS Elemental MediaTailor une anúncios em conteúdo ao vivo ou de vídeo sob demanda (VOD) substituindo ou inserindo anúncios no manifesto de origem. A inserção ou substituição dos anúncios depende de como os intervalos publicitários são configurados no manifesto de origem e se o conteúdo é VOD ou ao vivo. Um intervalo publicitário é o período durante a programação em que os comerciais são exibidos, enquanto os anúncios disponíveis são as unidades específicas do tempo de publicidade em um intervalo publicitário que podem ser preenchidas com anúncios.

- Com a substituição de anúncios, MediaTailor substitui segmentos de conteúdo por anúncios.
- Com a inserção de anúncios, MediaTailor insere o conteúdo do anúncio onde os segmentos não existem.

Para obter informações sobre como MediaTailor agrupar anúncios em conteúdo ao vivo e VOD, selecione o tópico aplicável.

Tópicos

- [Comportamento de costura de anúncios para VOD](#)
- [Comportamento de costura de anúncios ao vivo](#)

Comportamento de costura de anúncios para VOD

MediaTailor insere ou substitui anúncios no conteúdo VOD com base em como os marcadores de anúncios são configurados no manifesto de origem e se o servidor de decisão de anúncios (ADS) envia respostas do VMAP.

Para saber o comportamento de anúncios por configuração de marcador, consulte as seções a seguir.

Se os marcadores de anúncios estiverem presentes

AWS Elemental MediaTailor insere anúncios em que os marcadores de anúncios SCTE-35 estão presentes no manifesto de origem. Marcadores de anúncios com um EXT-X-CUE-OUT valor de 0 duração indicam a inserção do anúncio.

Diretrizes de HLS e marcadores

Siga estas diretrizes para sinalização SCTE pós-rolagem e ad pod:

Anúncios pré-lançados

Para pós-exibições HLS, os marcadores CUE-OUT/IN devem preceder o último segmento de conteúdo. Isso porque a especificação HLS exige a declaração específica de decoradores de tag antes de um segmento.

Por exemplo, considere a seguinte declaração.

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

AWS Elemental MediaTailor insere um pós-rola como o seguinte.

```
#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: Adicionar pods

CUE-OUT/IN as tags devem ser anexadas explicitamente a um segmento. Não use várias tags CUE-OUT/IN em sucessão para imitar o comportamento de um conjunto de anúncios.

Por exemplo, a declaração a seguir é um uso válido de CUE-OUT/IN para retratar um pod de anúncios.

```
#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
```

A declaração anterior acarreta um resultado semelhante ao seguinte.

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

A declaração a seguir é inválida.

```
#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
```

Se nenhum marcador de anúncio estiver presente

Os marcadores de anúncios são a forma recomendada de sinalizar quebras de anúncios em um manifesto. No entanto, os marcadores de anúncios não são obrigatórios. Se o manifesto não contiver

marcadores de anúncios para DASH ou HLS, MediaTailor faça uma única chamada para o ADS e crie intervalos publicitários com base na resposta:

- Se o ADS enviar uma resposta VAST, MediaTailor insere todos os anúncios da resposta em um intervalo publicitário no início do manifesto. Este é um pre-roll.
- Se o ADS enviar uma resposta do VMAP, MediaTailor usará os deslocamentos de tempo de intervalo do anúncio para criar intervalos e inseri-los em todo o manifesto nos horários especificados (pré-lançamento, meio ou pós-lançamento). MediaTailor usa todos os anúncios de cada intervalo de anúncio na resposta do VMAP para cada intervalo de anúncio no manifesto.

Note

Quando um segmento sobrepõe um ponto de inserção com VMAP para o conteúdo VOD, o MediaTailor arredonda para o ponto de inserção mais próximo.

Tip

Caso você deseje criar intervalos comerciais intermediários, mas o ADS não ofereça suporte a VMAP, verifique se há marcadores de anúncios no manifesto. O MediaTailor insere anúncios nos marcadores, conforme descrito nas seções a seguir.

Note

Para métodos de inserção de anúncios guiados pelo servidor, MediaTailor insere anúncios pré-lançados na parte superior do manifesto e o player os reproduz antes de outros tipos de anúncio.

Comportamento de costura de anúncios ao vivo

Nas transmissões ao vivo, AWS Elemental MediaTailor sempre realiza a substituição do anúncio, preservando o tempo total entre os marcadores do anúncio da forma mais próxima possível. Quando os marcadores de anúncio incluem o DURATION atributo, MediaTailor usa o valor para determinar a duração do intervalo publicitário. Cada CUE-OUT indicador deve ter uma duração ou um CUE-IN indicador correspondente em fluxos de trabalho ativos.

MediaTailor realiza a substituição de anúncios para conteúdo ao vivo do HLS e do DASH. Para obter informações sobre como MediaTailor calcular o posicionamento e o tempo dos intervalos publicitários, consulte [the section called “Ad markers \(Marcadores de anúncio\)”](#) e [the section called “Ad markers \(Marcadores de anúncio\)”](#)

Seleção e substituição de anúncios

AWS Elemental MediaTailor inclui anúncios da resposta VAST do servidor de decisão de anúncios (ADS) da seguinte forma:

- Se uma duração for especificada, MediaTailor seleciona um conjunto de anúncios que se encaixa na duração e os inclui.
- Se nenhuma duração for especificada, MediaTailor reproduz o máximo de anúncios possível até encontrar um marcador de anúncio que indique um retorno ao conteúdo principal.

AWS Elemental MediaTailor segue as seguintes diretrizes durante a substituição de anúncios ao vivo:

- MediaTailor tenta reproduzir anúncios completos, sem recortes ou truncagens.
- Sempre que MediaTailor encontra um marcador de anúncio que indica o fim do intervalo publicitário, ele retorna ao conteúdo subjacente. Isso pode significar a redução um anúncio em reprodução no momento.
- Ao final da duração, MediaTailor retorna ao conteúdo subjacente.
- Se MediaTailor ficar sem anúncios para reproduzir durante um intervalo publicitário, ele será reproduzido, se houver um configurado, ou retomará a reprodução do fluxo de conteúdo subjacente. Em geral, isso acontece quando não há anúncios transcodificados suficientes para preencher o intervalo comercial.

Tip

É possível definir o limite de tempo de anúncios não preenchido permitido em um intervalo com a configuração de limite de personalização. Para obter mais informações, consulte a [PlaybackConfiguration referência](#).

Pré-inscrição ao vivo para inserção de anúncios guiada pelo servidor

A pré-rolagem ao vivo funciona de forma diferente para métodos de inserção de anúncios guiados pelo servidor em comparação com a inserção de anúncios no servidor:

Inserção de anúncios no lado do servidor (modo costurado)

Os anúncios pré-lançados substituem parte do conteúdo ao vivo no início da sessão de cada espectador. Cada espectador vê a pré-inscrição em momentos diferentes, com base em quando entra na transmissão.

Métodos de inserção de anúncios guiados pelo servidor

MediaTailor coloca uma tag de intervalo de datas de pré-rolagem na parte superior de todos os manifestos de mídia com atributos. `CUE="PRE , ONCE"` Isso faz com que os jogadores solicitem e reproduzam anúncios pré-lançados uma vez no início da reprodução, apesar de compartilharem o mesmo manifesto não personalizado.

Requisitos de configuração:

- Servidor de decisão de anúncios pré-lançados ao vivo: configure um endpoint VAST para anúncios pré-lançados (pode ser diferente dos anúncios intermediários)
- Duração máxima permitida para pré-lançamento ao vivo: defina a duração máxima para anúncios pré-lançados (opcional - se omitidos, todos os anúncios retornados serão usados)

Implementação técnica:

- Usos `START-DATE="1970-01-01T00:00:00.000Z"` da tag de intervalo de datas de pré-rolagem (época do Unix)
- As solicitações da lista de ativos para pré-inscrição usam o servidor de decisão de anúncios pré-lançado configurado, em vez do ADS normal.
- Os jogadores identificam as solicitações de pré-inscrição por meio dos dados da `availId="aws-mediatailor-preroll-1"` lista de ativos.

 Important

Para transmissões ao vivo, os anúncios pré-rolados abordam o conteúdo em vez de atrasá-lo. Versões futuras podem suportar o modo de atraso de conteúdo por meio de opções adicionais de configuração.

 Note

O comportamento da pré-inscrição varia entre conteúdo ao vivo e VOD para inserção de anúncios guiada pelo servidor. O conteúdo ao vivo exige uma configuração explícita de pré-lançamento, enquanto o conteúdo VOD inclui a pré-inscrição por padrão usando o servidor de decisão de anúncios normal.

Exemplos

- Se o intervalo publicitário tiver uma duração definida para 70 segundos e a resposta do ADS contiver dois anúncios de 40 segundos, AWS Elemental MediaTailor reproduzirá um dos anúncios de 40 segundos. No tempo restante, ele alterna para a barreira configurada ou o conteúdo subjacente. A qualquer momento durante esse processo, caso encontre um indicador da deixa de entrada, o MediaTailor vai imediatamente para o conteúdo subjacente.
- Se o intervalo publicitário tiver uma duração definida como 30 segundos e o anúncio mais curto fornecido pela resposta do ADS for de 40 segundos, não MediaTailor reproduzirá anúncios. Se uma lista de anúncios estiver configurada, MediaTailor reproduza-a por 30 segundos ou até encontrar um indicador de entrada. Caso contrário, MediaTailor reproduz o conteúdo subjacente.

MediaTailor visão geral e implementação da inserção de anúncios guiada pelo servidor

AWS Elemental MediaTailor a inserção de anúncios guiada pelo servidor (SGAI) fornece uma alternativa à inserção de anúncios no lado do servidor, referenciando anúncios como playlists separadas, em vez de agrupá-los diretamente em playlists de mídia. Essa abordagem melhora o desempenho por meio de manifestos armazenáveis em cache e permite melhor escalabilidade.

Para obter informações sobre como usar a inserção de anúncios guiada pelo servidor com MediaTailor, escolha o tópico aplicável na lista a seguir.

Ativar na configuração de reprodução

Para permitir que os jogadores usem a inserção de anúncios guiada pelo servidor, você deve definir como `PLAYER_SELECT` na configuração de `Insertion Mode` reprodução. MediaTailor Isso permite que os jogadores selecionem a inserção de anúncios costurados ou guiados no momento da inicialização da sessão.

Crie uma sessão guiada pelo servidor

Ao criar sessões de reprodução, escolha o modo guiado. A maneira de fazer isso depende se seus jogadores usam sessões implícitas ou explícitas.

Sessões guiadas pelo servidor criadas implicitamente

Anexe `aws.insertionMode=GUIDED` à solicitação de playlist multivariante do HLS. Exemplo:

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

Em que:

- `playlist-endpoint` é o endpoint de reprodução exclusivo gerado pelo AWS Elemental MediaTailor quando a configuração foi criada.

Exemplo

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

- `hashed-account-id` é o seu Conta da AWS ID.

Exemplo

```
777788889999
```

- `origin-id` é o nome dado quando se cria a configuração.

Exemplo


```
myOrigin
```

- `index.m3u8` é o nome do manifesto do fluxo de teste mais sua extensão de arquivo. Defina isso para que você obtenha um manifesto totalmente identificado ao anexá-lo à origem do conteúdo do vídeo configurada em [the section called “Etapa 4: Criar uma configuração”](#).

Usando os valores dos exemplos anteriores, os valores completos URLs são os seguintes.

- Exemplo:

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

Sessões guiadas pelo servidor criadas explicitamente

Adicione `insertionMode=GUIDED` aos metadados JSON que o player envia no HTTP POST para o endpoint do prefixo de MediaTailor inicialização da sessão da configuração.

O exemplo a seguir mostra a estrutura dos metadados JSON:

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

Com esses metadados de inicialização, a sessão de reprodução usará a inserção de anúncios guiada pelo servidor.

Rastreamento de anúncios com o SGAI

O SGAI oferece suporte ao rastreamento de anúncios do lado do servidor e do lado do cliente. Você define o modo de relatório na inicialização da sessão. O modo não pode mudar durante a sessão.

Rastreamento do lado do servidor (padrão)

MediaTailor dispara sinais VAST automaticamente quando o jogador solicita segmentos de anúncios. O anúncio URIs na lista de ativos contém metadados de beacon criptografados

(awsBeaconData,,awsBeaconDomain). awsConfigurationName O jogador deve suportar a substituição de #EXT-X-DEFINE:QUERYPARAM variáveis HLS. A resposta da lista de ativos não inclui uma TRACKING seção.

Para obter detalhes sobre como o beaconing do lado do servidor SGAI funciona, consulte.

[Server-side rastreamento com inserção de anúncios guiada pelo servidor \(SGAI\)](#)

Rastreamento do lado do cliente

Adicione `aws.reportingMode=CLIENT` à sua solicitação de inicialização da sessão. A resposta da lista de ativos inclui uma TRACKING seção com um farol URLs que o player aciona durante a reprodução do anúncio. O endpoint `GetTracking` da API não é usado para sessões do SGAI. Em vez disso, cada resposta da lista de ativos inclui dados de rastreamento diretamente. Os dados de rastreamento usam o mesmo esquema JSON da resposta de rastreamento de inserção de anúncios no lado do servidor (SSAI).

Para obter detalhes, consulte [Server-guided inserção de anúncios](#).

Pré-busca guiada com batimento cardíaco manifesto

Para transmissões ao vivo do SGAI, você pode ativar a pré-busca de anúncios baseada em manifestos adicionando `aws.guidedPrefetchMode=MANIFEST` à sua solicitação de inicialização de sessão:

```
https://7777888899999.mediatailor.us-east-1.amazonaws.com/v1/master/7777888899999/myOrigin/index.m3u8?aws.insertionMode=GUIDED&aws.guidedPrefetchMode=MANIFEST
```

Quando ativado, MediaTailor anexa um identificador de sessão (`?aws.sessionId=<id>`) como parâmetro de consulta a cada URL de manifesto de mídia intersticial (`/v1/i-media`) na playlist multivariante. Sempre que o player atualiza um manifesto i-media, a solicitação chega MediaTailor com o ID da sessão, que é MediaTailor usado para enfileirar solicitações de busca prévia para os próximos intervalos publicitários.

- O `aws.guidedPrefetchMode` parâmetro aceita dois valores: MANIFEST (ativado) e OFF (desativado, padrão).
- O modo de pré-busca guiada só é válido para sessões do SGAI. Usá-lo com sessões costuradas retorna um erro.
- O DASH ainda não oferece suporte ao modo de pré-busca guiada.

- A pré-busca guiada é independente do modo de relatório — os beacons são acionados no momento da reprodução, não no momento da pré-busca.
- Não armazene em cache os manifestos i-media em sua CDN ao usar a pré-busca guiada. O mecanismo de pré-busca depende das solicitações de atualização do manifesto do jogador chegarem diretamente. MediaTailor Se sua CDN armazena /v1/i-media respostas em cache, MediaTailor não recebe as solicitações de pulsação e não pode acionar a pré-busca.

MediaTailor matriz de compatibilidade de recursos de inserção de anúncios guiada por servidor

AWS Elemental MediaTailor oferece dois métodos de inserção de anúncios com compatibilidade de recursos diferentes. A inserção de anúncios guiada pelo servidor funciona de forma diferente da inserção de anúncios no servidor, o que afeta a compatibilidade com alguns recursos. MediaTailor Use essa tabela para entender quais recursos funcionam com cada método de inserção de anúncios.

Compatibilidade de recursos por método de inserção de anúncios

Recurso	Inserção de anúncios no lado do servidor (SSAI)	Inserção de anúncios guiada pelo servidor (SGAI)
Pré-busca de anúncios	✓ Suportado	Sem suporte no momento
Supressão de anúncios	✓ Suportado	Não aplicável
Pré-lançamento e comportamento	Controlado pela MediaTailor configuração	Controlado pela MediaTailor configuração
Rastreamento de anúncios do lado do cliente	Usa GetTracking API	TRACKINGSeção de usos na lista de ativos (a GetTracking API não é usada)
Rastreamento de anúncios do lado do servidor	✓ Compatível — os beacons são acionados com base	✓ Compatível (somente HLS) — usa beaconing sem

Recurso	Inserção de anúncios no lado do servidor (SSAI)	Inserção de anúncios guiada pelo servidor (SGAI)
	em /v1/segment solicitações usando o ID da sessão	sessão com dados de beacon criptografados incorporados ao ad via. URIs #EXT-X-DEFINE:QUERYPARAM Requer HLS v11 ou posterior. O DASH ainda não é suportado.
Decoração de ID de anúncio	✓ Suportado	✗ Não compatível

Detalhes de compatibilidade

Pré-busca de anúncios

No momento, a pré-busca de anúncios não é suportada.

Supressão de anúncios

A supressão de anúncios não é compatível com os métodos de inserção de anúncios guiados pelo servidor porque os jogadores só buscam os anúncios que vão reproduzir.

Pré-lançamento e comportamento

O pré-lançamento e o tempo funcionam de forma diferente entre os métodos de inserção:

- Inserção de anúncios no servidor: MediaTailor controla quando os anúncios pré-lançados são reproduzidos com base nas configurações
- Inserção de anúncios guiada pelo servidor: MediaTailor insere anúncios pré-lançados na parte superior do manifesto. Seu player mostra esses anúncios primeiro e depois inicia seu conteúdo

Acompanhamento de anúncios

O rastreamento do lado do cliente usa mecanismos diferentes, dependendo do método de inserção do anúncio:

- Inserção de anúncios no lado do servidor (SSAI): usa o endpoint da API `GetTracking`
- Inserção de anúncios guiada pelo servidor (SGAI): MediaTailor fornece informações de rastreamento na `TRACKING` seção de cada resposta da lista de ativos. O endpoint `GetTracking` da API não é usado. A resposta de inicialização da sessão não inclui `umtrackingUrl`.

O rastreamento do lado do servidor também difere entre os métodos:

- Inserção de anúncios no lado do servidor (SSAI): MediaTailor dispara beacons quando o player busca segmentos de anúncios agrupados usando o ID da sessão. `/v1/segment/`
- Inserção de anúncios guiada pelo servidor (SGAI): usa beaconing sem sessão. MediaTailor incorpora dados de beacon criptografados (`awsBeaconData`, `awsBeaconDomain`, `awsConfigurationName`) no manifesto publicitário URLs que eles retornam na lista de ativos. O manifesto do anúncio usa `#EXT-X-DEFINE:QUERYPARAM` tags para que o player substitua esses valores em um segmento URLs. Quando o player solicita cada segmento do anúncio, MediaTailor decifra os dados, dispara o beacon apropriado e redireciona para o segmento de conteúdo. Quando os relatórios do lado do servidor estão ativos, MediaTailor omite a `TRACKING` seção da resposta da lista de ativos. Para obter detalhes, consulte [Server-side rastreamento com inserção de anúncios guiada pelo servidor \(SGAI\)](#).

Decoração de ID de anúncio

A decoração do Ad-ID não é compatível com a inserção de anúncios guiada pelo servidor porque os campos que preenchem os `X-AD-CREATIVE-SIGNALING` cabeçalhos só são conhecidos quando a lista de ativos é buscada, não quando o manifesto é escrito.

MediaTailor configuração de inserção de anúncios guiada por servidor para transmissões ao vivo

AWS Elemental MediaTailor a inserção de anúncios guiada pelo servidor para conteúdo ao vivo oferece benefícios significativos de desempenho por meio de manifestos que podem ser armazenados em cache. A configuração do SGAi para conteúdo ao vivo usa os mesmos parâmetros

principais do VOD, com considerações específicas para as características da transmissão ao vivo e processamento em tempo real.

Requisitos para SGAI ao vivo

Antes de ativar o SGAI para conteúdo ao vivo, verifique se você tem o seguinte:

- Sua transmissão ao vivo inclui marcadores DATERANGE formatados corretamente
- As durações dos intervalos publicitários são consistentes e previsíveis
- Seu CDN está configurado para armazenar em cache os manifestos SGAI de forma adequada
- Os jogadores oferecem suporte a fluxos de trabalho de inserção de anúncios guiados pelo servidor
- Seu servidor de decisão de anúncios pode lidar com solicitações em tempo real de conteúdo ao vivo

Requisitos do jogador

Os jogadores devem estar configurados para lidar adequadamente com os manifestos ao vivo do SGAI:

- Support para fluxos de trabalho de inserção de anúncios guiados pelo servidor
- Capacidade de processar orientações de inserção de anúncios a partir de manifestos
- Tratamento adequado do tempo e sincronização da transmissão ao vivo
- Para conteúdo HLS: Support para HLS versão 8 e EXT-X-DATERANGE com o atributo CLASS. Versão 11 para beaconing do lado do servidor.
- Para conteúdo HLS: suporte à substituição de EXT-X-DEFINE variáveis

Configuração de reprodução ao vivo

Para habilitar o SGAI para conteúdo ao vivo, crie uma configuração de reprodução com as seguintes configurações:

Example Configuração de reprodução ao vivo do SGAI

```
{  
  "Name": "LiveSGAIConfig",
```

```
"VideoContentSourceUrl": "https://your-live-origin.com/live/stream.m3u8",  
"AdDecisionServerUrl": "https://your-ads.com/ads",  
"PersonalizationThresholdSeconds": 1,  
"InsertionMode": "PLAYER_SELECT"  
}
```

A seguir estão as principais considerações para a configuração ativa do SGAI:

VideoContentSourceUrl

Deve apontar para um fluxo HLS ao vivo com marcadores SCTE-35 DATERANGE formatados corretamente. O stream deve manter durações de segmentos e variantes de taxa de bits consistentes.

Solicitações de manifesto ao vivo do SG

Os manifestos ativos do SGAI usam o mesmo padrão de URL da inserção tradicional de anúncios:

```
https://your-config.mediatailor.region.amazonaws.com/v1/master/config-name/  
manifest.m3u8?aws.insertionMode=GUIDED
```

Pré-busca baseada em manifesto para SGAI ao vivo

Para fluxos de trabalho SGAI ativos, você pode ativar o heartbeating de pré-busca baseado em manifesto para reduzir a latência de preenchimento de anúncios. Adicione `aws.guidedPrefetchMode=MANIFEST` à solicitação do manifesto:

```
https://your-config.mediatailor.region.amazonaws.com/v1/master/config-name/  
manifest.m3u8?aws.insertionMode=GUIDED&aws.guidedPrefetchMode=MANIFEST
```

Quando ativado, MediaTailor anexa um identificador de sessão (`?aws.sessionId=<id>`) como parâmetro de consulta a cada URL de manifesto de mídia intersticial (`/v1/i-media`) na playlist multivariante. Sempre que o player atualiza um manifesto i-media, a solicitação chega MediaTailor com o ID da sessão, que é MediaTailor usado para identificar a sessão e enfileirar solicitações de pré-busca para os próximos intervalos publicitários.

⚠ Important

Não armazene em cache os manifestos i-media em sua CDN ao usar a pré-busca guiada. O mecanismo de pulsação prévia depende das solicitações de atualização do manifesto do jogador chegarem diretamente. MediaTailor Se sua CDN armazena em cache e fornece `/v1/i-media` respostas, MediaTailor não recebe as solicitações de pulsação e não pode acionar a pré-busca. Configure sua CDN para transmitir `/v1/i-media/*` solicitações até MediaTailor quando `aws.guidedPrefetchMode=MANIFEST` estiver em uso.

A pré-busca guiada é independente do modo de relatório. Se você usa o rastreamento do lado do servidor (padrão) ou do lado do cliente (`aws.reportingMode=CLIENT`), os beacons são acionados no momento da reprodução, não quando os anúncios são pré-buscados. Para obter informações gerais sobre como a pré-busca de anúncios funciona em MediaTailor. [Pré-busca de anúncios](#)

Testando a configuração ativa do SGAI

Verifique sua configuração ao vivo do SGAI com estas etapas de validação:

1. Geração de manifesto de teste

Solicite o URL do manifesto ativo do SGAI e verifique se ele retorna conteúdo armazenável em cache com a orientação adequada de inserção de anúncios.

2. Verifique o cache do CDN

Verifique se sua CDN está armazenando em cache os manifestos SGAI de acordo com os valores TTL configurados.

3. Teste e inserção

Confirme se os jogadores conseguem inserir anúncios com sucesso usando as orientações fornecidas nos manifestos do SGAI.

4. Monitore o desempenho

Use CloudWatch métricas para verificar a redução da carga de origem e as melhores taxas de acerto do cache.

MediaTailor configuração de inserção de anúncios guiada pelo servidor para conteúdo VOD

AWS Elemental MediaTailor a inserção de anúncios guiada por servidor para conteúdo VOD oferece benefícios significativos de desempenho por meio de manifestos altamente armazenáveis em cache e processamento reduzido do servidor. A configuração do SGAI para conteúdo VOD aproveita a natureza estática dos video-on-demand ativos para maximizar a eficiência do armazenamento em cache e minimizar as solicitações de origem, tornando-o ideal para grandes bibliotecas de conteúdo com padrões de visualização repetidos.

Requisitos para VOD SGAI

Antes de habilitar o SGAI para conteúdo VOD, verifique se você tem o seguinte:

- Seu conteúdo VOD inclui marcadores de anúncios formatados corretamente (SCTE-35 ou metadados cronometrados)
- O conteúdo é armazenado em uma origem confiável com disponibilidade consistente
- Seu CDN está configurado para armazenar em cache os manifestos SGAI com os valores TTL apropriados
- Os jogadores oferecem suporte a fluxos de trabalho de inserção de anúncios guiados pelo servidor
- Seu servidor de decisão de anúncios pode lidar com metadados e segmentação específicos de VOD

Requisitos do jogador

Os jogadores devem estar configurados para lidar com manifestos VOD do SGAI e inserção de anúncios:

- Support para fluxos de trabalho de inserção de anúncios guiados pelo servidor
- Capacidade de processar orientações de inserção de anúncios a partir de manifestos de VOD
- Support para inserção de anúncios do lado do cliente durante a reprodução de VOD
- Tratamento adequado das operações de busca em intervalos publicitários
- Support para duração do conteúdo e rastreamento de posição

Configuração de reprodução de VOD

Para habilitar o SGAI para conteúdo VOD, crie uma configuração de reprodução com as seguintes configurações:

Example Configuração de reprodução SGAI VOD

```
{
  "Name": "VODSGAIConfig",
  "VideoContentSourceUrl": "https://your-vod-origin.com/content/",
  "AdDecisionServerUrl": "https://your-ads.com/ads",
  "PersonalizationThresholdSeconds": 5,
  "InsertionMode": "PLAYER_SELECT"
}
```

A seguir estão as principais considerações para a configuração do VOD SGAI:

VideoContentSourceUrl

Deve apontar para sua biblioteca de conteúdo VOD com padrões de URL consistentes. Garanta que a origem possa lidar com o volume esperado de solicitações e forneça uma entrega confiável de conteúdo.

ConfigurationAliases

Inclua parâmetros específicos do VOD, como duração do conteúdo, gênero ou informações da série, que podem ser usados para segmentação de anúncios sem afetar a capacidade de armazenamento em cache do manifesto.

ManifestProcessingRules

Ative a passagem do marcador de anúncios para preservar as informações de tempo do conteúdo original, o que é especialmente importante para conteúdo VOD com posições predefinidas de interrupção do anúncio.

Solicitações de manifesto SGAI VOD

Os manifestos VOD do SGAI usam o mesmo padrão de URL (inserção tradicional de anúncios VOD).

```
https://your-config.mediatailor.region.amazonaws.com/v1/master/config-name/content-path/manifest.m3u8?aws.insertionMode=GUIDED
```

Segmentação de anúncios específica para VOD

O conteúdo VOD oferece oportunidades exclusivas de segmentação de anúncios:

Segmentação por metadados de conteúdo

Aproveite os metadados de conteúdo VOD para melhorar a segmentação de anúncios:

- Gênero e categoria: segmente anúncios com base no tipo de conteúdo (drama, comédia, documentário)
- Classificação do conteúdo: garanta um conteúdo de anúncio adequado à idade (classificações G, PG, R)
- Séries e temporadas: segmente anúncios para continuidade da série ou conteúdo relacionado
- Data de lançamento: meta com base na idade do conteúdo (novos lançamentos versus conteúdo do catálogo)
- Duração do conteúdo: ajuste a carga do anúncio com base no tamanho total do conteúdo

Visualizando a segmentação por contexto

Os padrões de visualização de VOD permitem a segmentação contextual de anúncios:

- Hora do dia: segmente anúncios com base em quando o conteúdo está sendo assistido
- Assistir compulsivamente: ajuste a frequência dos anúncios para usuários que assistem a vários episódios
- Taxa de conclusão: meta com base nos padrões históricos de conclusão do conteúdo do usuário
- Tipo de dispositivo: otimize os formatos de anúncio para o dispositivo de visualização (TV, celular, tablet)

Testando a configuração do SGAI VOD

Verifique a configuração do SGAI VOD com estas etapas de validação:

1. Geração de manifesto de teste

Solicite o manifesto SGAI VOD URLs para diferentes tipos de conteúdo e verifique se eles retornam conteúdo armazenável em cache com a orientação adequada de inserção de anúncios.

2. Verifique o cache do CDN

Verifique se sua CDN está armazenando em cache os manifestos SGAI de acordo com os valores de TTL configurados e alcançando altas taxas de acerto do cache.

3. Teste e inserção

Confirme se os jogadores conseguem inserir anúncios com sucesso usando as orientações fornecidas nos manifestos do SGAI para vários conteúdos de VOD.

4. Operações de busca de testes

Verifique se a busca no conteúdo VOD funciona corretamente em intervalos publicitários e mantém a posição de reprodução adequada.

5. Monitore o desempenho

Use CloudWatch métricas para verificar a redução da carga de origem, melhores taxas de acerto do cache e taxas de inserção de anúncios bem-sucedidas.

Principais cenários de teste

Teste esses cenários específicos de VOD:

- Conteúdo popular: verifique as altas taxas de acerto do cache para ativos de VOD acessados com frequência
- Conteúdo de formato longo: teste a inserção de anúncios em filmes ou episódios longos com vários intervalos publicitários
- Conteúdo da série: verifique a segmentação consistente de anúncios em todos os episódios de uma série
- Gêneros diferentes: teste a segmentação de anúncios com base nos metadados e no gênero do conteúdo

Melhores práticas de otimização do VOD SGAI

Otimize sua implementação do SGAI VOD para obter o máximo desempenho:

Otimização de cache

- Maximize os valores de TTL: use durações de cache mais longas para manifestos de VOD, pois o conteúdo não muda
- Minimize as chaves de cache: reduza as variações das chaves de cache para melhorar as taxas de acerto
- Pré-aqueça conteúdo popular: manifestos de cache para conteúdo VOD popular ou em destaque
- Monitore o desempenho do cache: acompanhe as taxas de acerto do cache e otimize com base nos padrões de uso

Otimização da entrega de conteúdo

- Padrões de URL consistentes: use estruturas de URL previsíveis para melhorar o armazenamento em cache
- Padronização de metadados: garanta metadados de conteúdo consistentes para uma segmentação confiável de anúncios
- Posicionamento do intervalo do anúncio: otimize o posicionamento do intervalo do anúncio para transições naturais de conteúdo
- Variantes de qualidade: garanta que o SGAI funcione em todas as variantes de taxa de bits do seu conteúdo VOD

MediaTailor requisitos de integração do servidor de anúncios

Para integrar seu servidor de anúncios AWS Elemental MediaTailor, seu servidor de anúncios deve enviar um XML que esteja em conformidade com as especificações do IAB para as versões compatíveis do VAST e do VMAP. Use um validador VAST público para garantir que as tags estejam bem formadas.

AWS Elemental MediaTailor suporta respostas VAST e VMAP de servidores de decisão de anúncios. AWS Elemental MediaTailor também suporta o proxy de metadados VPAID por meio de nossa API de relatórios do lado do cliente para inserção de anúncios do lado do cliente. Para obter informações sobre relatórios no lado do cliente, consulte [Client-side rastreamento de anúncios](#).

MediaTailor suporta as seguintes versões do VAST, VMAP e VPAID:

- Até [VAST 4.3](#)

MediaTailor aceita versões de resposta por meio do VAST 4.3, mas alguns recursos avançados do VAST 4.0 e superior não são suportados.

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

Requisitos VAST

A resposta VAST do servidor de anúncios deve conter elementos `TrackingEvents` compatíveis com IAB e tipos de evento padrão, como `impression`. Se você incluir eventos de rastreamento não padrão, AWS Elemental MediaTailor rejeitará a resposta do VAST e não fornecerá um anúncio disponível.

O VAST 3.0 introduziu suporte para conjuntos de anúncios, que é a entrega de um conjunto de anúncios lineares sequenciais. Se um anúncio específico em um pod de anúncios não estiver disponível, AWS Elemental MediaTailor registra um erro no CloudWatch registro de interações do ADS. Em seguida, ele tenta inserir o próximo anúncio no conjunto. Dessa forma, MediaTailor percorre os anúncios no pod até encontrar um que possa ser usado.

Definição de destino

Para segmentar jogadores específicos para seus anúncios, você pode criar modelos para suas tags de anúncio URLs e. Para obter mais informações, consulte [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#).

AWS Elemental MediaTailor usa proxy para o player `user-agent` e os `x-forwarded-for` cabeçalhos quando envia a solicitação VAST do servidor de anúncios e quando faz as chamadas de rastreamento do lado do servidor. Verifique se o servidor de anúncios consegue processar esses cabeçalhos. Também use `[session.user_agent]` ou `[session.client_ip]` e passe esses valores em strings de consulta na tag e no URL do anúncio. Para obter mais informações, consulte [MediaTailor variáveis de sessão para solicitações de ADS](#).

Adicionar chamadas

AWS Elemental MediaTailor chama seu URL de anúncios VAST conforme definido em sua configuração. Ele substitui quaisquer parâmetros específicos do jogador ou da sessão ao fazer a chamada do anúncio. MediaTailor segue até sete níveis de invólucros e redirecionamentos VAST na resposta VAST. Em cenários de transmissão ao vivo, MediaTailor faz chamadas publicitárias simultaneamente no início da disponibilidade do anúncio para jogadores conectados. Na prática,

por causa da oscilação, essas chamadas de anúncios podem ser distribuídas ao longo de alguns segundos. Certifique-se de que seu servidor de anúncios possa lidar com o número de conexões simultâneas que esse tipo de chamada exige. MediaTailor suporta a pré-busca de respostas VAST para fluxos de trabalho ativos. Para obter mais informações, consulte [Pré-busca de anúncios](#).

Manipulação criativa

Quando AWS Elemental MediaTailor recebe a resposta do ADS VAST, para cada criativo, ele identifica a taxa de bits mais alta MediaFile para transcodificação e a usa como fonte. Ele envia esse arquivo para o on-the-fly transcodificador para transformação em representações que se ajustem às taxas de bits e resoluções multivariantes da playlist do player. Para obter melhores resultados, certifique-se de que seu arquivo de mídia com maior taxa de bits seja um MP4 ativo de alta qualidade com predefinições de manifesto válidas. Quando as predefinições de manifesto não são válidas, as tarefas de transcodificação falham, resultando na não exibição de anúncios. Exemplos de predefinições que não são válidas incluem formatos de arquivo de entrada não suportados, como, e determinadas especificações de representação ProRes, como a resolução 855X481.

Para obter uma lista de formatos compatíveis para entradas de arquivos de mídia, consulte a MP4linha de [formatos de entrada compatíveis](#) no Guia do AWS Elemental MediaConvert usuário.

Indexação criativa

AWS Elemental MediaTailor indexa exclusivamente cada criativo pelo valor do id atributo fornecido no elemento. <Creative> Se o ID do criativo não for especificado, MediaTailor usa o URL do arquivo de mídia para o índice.

A seguinte declaração de exemplo mostra o ID do criativo.

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

Se você definir seu próprio criativo IDs, use um ID novo e exclusivo para cada criativo. Não reutilize a criatividade IDs. AWS Elemental MediaTailor armazena conteúdo criativo para uso repetido e encontra cada um por seu ID indexado. Quando um novo criativo chega, o serviço primeiro verifica o ID em relação ao índice. Caso o ID esteja presente, o MediaTailor usa o conteúdo armazenado, em vez de reprocessar o conteúdo recebido. Se você reutilizar uma ID criativa, MediaTailor usará o anúncio antigo armazenado e não reproduzirá seu novo anúncio.

Extensões VAST fornecidas por parceiros de veiculação de anúncios

Para ajudar a evitar colisões com a criatividade IDs, você pode usar extensões fornecidas por parceiros de veiculação de anúncios para a resposta do VAST. MediaTailor suporta extensões de SpringServe, PubMoxer e FreeWheel. Quando você ativa as substituições da extensão VAST, MediaTailor substitui a ID criativa padrão pelo valor da extensão.

Para habilitar esse recurso, [envie um ticket de AWS Support](#) para solicitar que a criação baseada na extensão VAST IDs seja habilitada. Inclua as seguintes informações no ticket do Support:

- AWS Região
- AWS ID da conta
- MediaTailor nomes de configuração de reprodução

Para validar se os criativos baseados na extensão VAST IDs estão habilitados em sua conta, recomendamos que você também solicite que o RAW_ADS_RESPONSE registro seja ativado em uma configuração de teste ou teste de reprodução. Com o registro, você pode visualizar a resposta VAST original que o ADS recebe e confirmar se o criativo IDs correto foi usado.

Requisitos do VPAID

VPAID permite que os editores veiculem anúncios de vídeo altamente interativos e forneçam métricas de visibilidade sobre os streamings monetizados. Para obter informações sobre VPAID, consulte a especificação [VPAID](#).

AWS Elemental MediaTailor suporta uma combinação de anúncios MP4 lineares server-side-stitched client-side-inserted VAST e criativos interativos VPAID na mesma disponibilidade de anúncios. Ele preserva a ordem em que eles aparecem na resposta VAST. MediaTailor segue os redirecionamentos do VPAID por meio de um máximo de sete níveis de invólucros. A resposta dos relatórios no lado do cliente inclui metadados VPAID desencapsulados.

Para usar VPAID, siga estas diretrizes:

- Configure uma MP4 lista para seus criativos VPAID. AWS Elemental MediaTailor preenche os espaços de anúncios VPAID com sua lista configurada e fornece metadados de anúncios VPAID para o player cliente usar para veicular os anúncios interativos. Caso você não tenha uma barreira configurada, quando um anúncio VPAID é exibido, o MediaTailor fornece os metadados do anúncio por meio de relatórios no lado do cliente normalmente. Ele também registra um erro CloudWatch sobre a lousa ausente. Para obter mais informações, consulte [MediaTailor ardósia e inserção](#) e [Criando uma MediaTailor configuração de reprodução](#).

- Use relatórios do lado do cliente. AWS Elemental MediaTailor suporta VPAID por meio de nossa API de relatórios do lado do cliente. Para obter mais informações, consulte [Client-side rastreamento de anúncios](#).

Teoricamente, é possível usar o modo de relatórios no lado do servidor padrão com VPAID. No entanto, caso use relatórios no lado do servidor, você perde todas as informações sobre a presença do anúncio VPAID e dos metadados próximos, porque eles só estão disponíveis por meio da API no lado do cliente.

- Em cenários ao vivo, certifique-se de que seus anúncios, indicados por `EXT-X-CUE-OUT: Duration`, sejam longos o suficiente para acomodar qualquer interatividade do usuário no VPAID. Por exemplo, se o XML VAST especificar um anúncio VPAID com 30 segundos de duração, considere configurar a disponibilidade do anúncio para mais de 30 segundos. Esse tempo adicional dá aos usuários mais chances de interagir com o anúncio. Se você não adicionar tempo, poderá perder os metadados VPAID porque a duração restante na disponibilidade do anúncio não é longa o suficiente para acomodar o anúncio VPAID.

MediaTailor gerenciamento de configuração de reprodução

Esta seção aborda as principais tarefas para gerenciar as configurações de AWS Elemental MediaTailor reprodução. Você pode aprender a criar uma nova configuração para configurar fluxos de conteúdo e fornecer acesso a dispositivos de reprodução, visualizar detalhes de uma configuração existente, editar uma configuração para atualizar configurações, como servidores de origem e servidores de decisão de anúncios, e excluir uma configuração que não é mais necessária.

Tópicos

- [Criando uma MediaTailor configuração de reprodução](#)
- [Visualizando detalhes da MediaTailor configuração](#)
- [Editando MediaTailor as configurações](#)
- [Excluindo configurações MediaTailor](#)

Criando uma MediaTailor configuração de reprodução

Este tópico mostra como criar uma configuração para começar a receber fluxos de conteúdo. AWS Elemental MediaTailor Também mostra como fornecer um ponto de acesso para dispositivos de reprodução downstream solicitarem conteúdo.

Você pode usar o AWS Elemental MediaTailor console, o AWS Command Line Interface (AWS CLI) > ou a MediaTailor API para criar uma configuração. Para obter informações sobre como criar uma configuração por meio da MediaTailor API AWS CLI or, consulte a [Referência AWS Elemental MediaTailor da API](#).

Ao criar uma configuração, não coloque informações de identificação confidenciais em campos de formato livre, como o campo Nome da configuração. As informações de identificação podem incluir coisas como números de contas de clientes. Além disso, não use informações de identificação ao trabalhar no MediaTailor console, na API REST AWS CLI, no ou AWS SDKs. Todos os dados que você inserir MediaTailor podem ser coletados para inclusão nos registros de diagnóstico ou nos CloudWatch Eventos da Amazon.

Como adicionar uma configuração (console)

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. Na página Configurações, selecione Criar configuração.
3. Preencha os campos de configuração e configuração adicionais, conforme descrito nos tópicos a seguir:
 - [Configurações necessárias](#)
 - [Configurações opcionais](#)
4. Escolha Criar configuração.

AWS Elemental MediaTailor exibe a nova configuração na tabela da página Configurações.

5. (Recomendado) Configure uma CDN AWS Elemental MediaTailor para solicitações de manifestos e relatórios. Você pode usar a reprodução da configuração URLs para a configuração do CDN. Para obter informações sobre como configurar uma CDN para solicitações de manifestos e relatórios, consulte [Usando uma CDN para otimizar a personalização de MediaTailor anúncios e a entrega de conteúdo](#).

Configurações necessárias

Ao criar uma configuração, você deve incluir as configurações necessárias a seguir.

Nome

Insira um nome exclusivo que descreva a configuração. O nome é o principal identificador da configuração. O tamanho máximo permitido é de 512 caracteres.

Fonte de conteúdo

Insira o prefixo do URL do manifesto para esse streaming, menos o ID do ativo. O tamanho máximo é de 512 caracteres.

Por exemplo, o prefixo de URL `http://origin-server.com/a/` é válido para um URL de playlist multivariante HLS de `http://origin-server.com/a/main.m3u8` e para um URL DASH MPD de `http://origin-server.com/a/dash.mpd`. Também é possível inserir um prefixo mais curto, como `http://origin-server.com`, mas o `/a/` deve ser incluído no ID do ativo na solicitação do player para o conteúdo.

Note

Caso a origem do conteúdo use HTTPS, o certificado deve ser de uma autoridade de certificação bem conhecida. Não pode ser um certificado autoassinado. Se você usa um certificado autoassinado, AWS Elemental MediaTailor não consegue se conectar à origem do conteúdo e não pode veicular manifestos em resposta às solicitações dos jogadores.

Servidor de decisão de anúncios

Insira o URL do servidor de decisões de anúncios (ADS). Esse é o URL com variáveis conforme descrito em [Etapa 3: Configurar o URL de solicitação do ADS e os parâmetros de consulta](#) ou o URL VAST estático que você está usando para fins de teste. O tamanho máximo é de 25.000 caracteres.

Note

Caso o ADS use HTTPS, o certificado deve ser de uma autoridade de certificação bem conhecida. Não pode ser um certificado autoassinado. O mesmo também se aplica ao anúncio de mezanino URLs retornado pelo ADS. Se você usar um certificado autoassinado, não AWS Elemental MediaTailor poderá recuperar e incluir anúncios nos manifestos a partir da origem do conteúdo.

Configurações opcionais

Opcionalmente, você pode definir aliases de configuração, detalhes de personalização e configurações avançadas no MediaTailor console, na MediaTailor API ou no AWS Command Line Interface (CLI).

Aliases de configuração

A seguir estão os aliases de configuração opcionais que você pode configurar no MediaTailor console ou com a MediaTailor API.

Variável de parâmetro do jogador

Para configuração dinâmica do domínio durante a inicialização da sessão, adicione uma ou mais variáveis de parâmetros do player.

Para obter mais informações sobre o uso de variáveis de parâmetros do player para configurar domínios dinamicamente, consulte [MediaTailor variáveis de domínio para várias fontes de conteúdo](#)

Configuração de log

A seguir estão as configurações de log.

Porcentagem ativada

Define a porcentagem de registros da sessão de configuração de reprodução que MediaTailor grava no CloudWatch Logs. Por exemplo, se sua configuração de reprodução tiver 1.000 sessões e você definir a porcentagem ativada como 60, MediaTailor grava 600 registros de sessão em CloudWatch Logs.

Quando você ativa essa opção, o MediaTailor cria automaticamente uma função vinculada ao serviço que permite ao MediaTailor gravar e gerenciar registros de sessão na sua conta do CloudWatch Logs. Para obter mais informações, consulte [Usando funções vinculadas a serviços para o MediaTailor](#).

Estratégias de registro

Indica o método usado para coletar os registros do MediaTailor emitidos. Para enviar registros diretamente para o CloudWatch Logs, escolha `LEGACY_CLOUDWATCH`. Para enviar registros para

o CloudWatch Logs, que então os vende para o destino de sua escolha, escolha `VENDED_LOGS`. Os destinos compatíveis são um grupo de CloudWatch registros de registros, um bucket do Amazon S3 e um stream do Amazon Data Firehose.

É necessária uma configuração adicional para troncos vendidos. Para configurar, consulte [Usando registros vendidos](#).

Eventos de aceitação do registro de interações do ADS

Indica que MediaTailor emitirá `RAW_ADS_RESPONSE` registros para sessões inicializadas com essa configuração.

O evento de `RAW_ADS_RESPONSE` log contém toda a resposta VAST ou VMAP do ADS. Dessa forma, os registros podem ser extensos e aumentar seus custos de registro.

Registro de interação do ADS exclui eventos

Indica que MediaTailor não emitirá os eventos selecionados nos registros que descrevem as interações com o ADS.

Para obter uma descrição dos eventos de registro do ADS, consulte [Registros do ADS](#).

Registro de interação do serviço manifesto, exclui eventos

Indica que MediaTailor não emitirá os eventos selecionados nos registros que descrevem as interações com o serviço de manifesto.

Para obter uma descrição dos eventos do log de serviço do manifesto, consulte [Registros de manifestos](#).

Condicionamento de anúncios

As informações a seguir determinam quais ações são MediaTailor tomadas para condicionar os anúncios antes de agrupá-los em um fluxo de conteúdo.

Condicionamento de arquivos de mídia de streaming

Determina a lógica MediaTailor usada ao decidir quais anúncios costurar.

- Quando o condicionamento do arquivo de mídia de streaming está definido como Transcodificar, MediaTailor transcodifica os arquivos de mídia com a *progressive* entrega e os une ao manifesto. Se não houver anúncios suficientes com arquivos de mídia

progressive de entrega para preencher a disponibilidade, MediaTailor transcodifica e usa esses anúncios com streaming entrega.

- Quando o condicionamento de arquivos de mídia de streaming está definido como Nenhum MediaTailor , agrupa anúncios com arquivos streaming de mídia de entrega no manifesto sem transcodificá-los. Se não houver anúncios suficientes com arquivos de mídia streaming de entrega para preencher a disponibilidade, MediaTailor transcodifica e usa esses anúncios com progressive entrega.

Detalhes de personalização

Veja a seguir os detalhes de personalização que você pode configurar no MediaTailor console ou com a MediaTailor API.

Anúncio de barreira

Insira o URL de um MP4 recurso de alta qualidade MediaTailor usado para preencher o tempo de anúncio não preenchido. A configuração do Slate é opcional para a maioria dos fluxos de trabalho, mas obrigatória para anúncios VPAID. Para obter informações abrangentes sobre o comportamento da ardósia, os requisitos de configuração e quando a ardósia é exibida, consulte.

[MediaTailor ardósia e inserção](#)

Para-choque de partida

O URL da localização inicial do ativo bumper. Os bumpers são pequenos cliques de vídeo ou áudio que são reproduzidos no início ou no final de um intervalo publicitário. Eles podem ser armazenados no S3 da Amazon ou em um serviço de armazenamento diferente. Para saber mais sobre pára-choques, consulte [MediaTailor pára-choques e inserção](#).

Pára-choque final

O URL da localização do ativo final do bumper. Os bumpers são pequenos cliques de vídeo ou áudio que são reproduzidos no início ou no final de um intervalo publicitário. Eles podem ser armazenados no S3 da Amazon ou em um serviço de armazenamento diferente. Para saber mais sobre pára-choques, consulte [MediaTailor pára-choques e inserção](#).

Limite de personalização

Só se aplica quando usado em conjunto com o anúncio em ardósia. Define a duração máxima do tempo de anúncio não preenchido (em segundos) permitido em um intervalo publicitário antes de MediaTailor abandonar a personalização e mostrar o conteúdo subjacente. Esse recurso se aplica à substituição de anúncios em streamings ao vivo e VOD e não à inserção de anúncios,

porque ele depende de um fluxo de conteúdo subjacente. Para cenários de comportamento detalhados e informações abrangentes, consulte [Como funciona o limite de personalização](#).

Servidor de decisões de anúncios pre-roll ao vivo

Para inserir anúncios no início de um streaming ao vivo antes do início da reprodução do conteúdo principal, insira o URL do anúncio pre-roll do servidor de decisões de anúncios (ADS). Esse é o URL com variáveis conforme descrito em [Etapa 3: Configurar o URL de solicitação do ADS e os parâmetros de consulta](#) ou o URL VAST estático que você está usando para fins de teste. O tamanho máximo é de 25.000 caracteres.

Note

Caso o ADS use HTTPS, o certificado deve ser de uma autoridade de certificação bem conhecida. Não pode ser um certificado autoassinado. O mesmo também se aplica ao anúncio de mezanino URLs retornado pelo ADS. Se você usar um certificado autoassinado, não AWS Elemental MediaTailor poderá recuperar e incluir anúncios nos manifestos a partir da origem do conteúdo.

Para obter informações sobre como funciona o pre-roll, consulte [MediaTailor pré-rolo e inserção](#).

Duração máxima permitida de pre-roll ao vivo

Ao inserir anúncios no início de um streaming ao vivo, insira a duração máxima permitida para a disponibilidade de anúncios pre-roll. O MediaTailor não excederá essa duração ao inserir anúncios. Se a resposta do ADS contiver mais anúncios do que cabe nessa duração, MediaTailor preencha o formulário com o maior número possível de anúncios, sem ultrapassar a duração. Para obter mais detalhes sobre como os MediaTailor preenchimentos são disponibilizados, consulte [Comportamento de costura de anúncios ao vivo](#).

Aproveite o modo de supressão

Define o modo de supressão de disponibilidade, também conhecido como supressão de anúncios. Por padrão, a supressão de anúncios está desativada e MediaTailor preenche tudo com anúncios ou ardósia. Quando o modo está definido como `BEHIND_LIVE_EDGE`, a supressão de anúncios está ativa e MediaTailor não preenche os intervalos de anúncios dentro ou após o tempo de valor da supressão de disponibilidade na janela de retrospectiva do manifesto. Quando o modo está definido como `AFTER_LIVE_EDGE`, a supressão de anúncios está ativa. MediaTailor não preenche intervalos publicitários durante ou após o período de supressão de

disponibilidade, que é a vantagem ativa mais o valor da supressão de disponibilidade mais o tempo de espera.

Valor de supressão de aproveitamento

O valor de supressão de disponibilidade é um tempo de compensação de borda em tempo real. HH:MM:SS MediaTailor desta vez, não preencherá intervalos publicitários anteriores ou posteriores na janela de retrospectiva do manifesto.

Modo de inserção

O Modo de Inserção controla se os jogadores podem usar a inserção de anúncios costurados ou guiados. O padrão, STITCHED_ONLY, força todas as sessões do jogador a usarem a inserção de anúncios costurados (do lado do servidor). InsertionMode A configuração para PLAYER_SELECT permite que os jogadores selecionem a inserção de anúncios costurados ou guiados no momento da inicialização da sessão. O padrão para jogadores que não especificam um modo de inserção é costurado.

Configurações avançadas

A seguir estão as configurações opcionais avançadas. Você pode configurá-los no MediaTailor console, com o AWS Command Line Interface (AWS CLI) ou usando a MediaTailor API.

Prefixo do segmento de conteúdo da CDN

Permite AWS Elemental MediaTailor criar manifestos em seu caminho URLs de CDN para segmentos de conteúdo. Antes de seguir essa etapa, configure uma regra na CDN para extrair segmentos do servidor de origem. Em CDN content segment prefix (Prefixo do segmento de conteúdo CDN), digite o caminho do prefixo CDN.

Para obter mais informações sobre a integração MediaTailor com uma CDN, consulte. [Usando uma CDN para otimizar a personalização de MediaTailor anúncios e a entrega de conteúdo](#)

Prefixo de segmento de anúncios da CDN

Permite AWS Elemental MediaTailor criar manifestos com URLs seu próprio caminho de CDN para segmentos de anúncios. Por padrão, MediaTailor veicula segmentos de anúncios de uma CloudFront distribuição interna da Amazon com configurações de cache padrão. Para preencher o campo CDN ad segment prefix (Prefixo do segmento de anúncios da CDN), você deve configurar uma regra na CDN para extrair segmentos de anúncio na seguinte origem, como o exemplo a seguir.


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

Em CDN ad segment prefix (Prefixo do segmento de anúncios da CDN), digite o nome do prefixo da CDN na configuração.

Para obter mais informações sobre a integração MediaTailor com uma CDN, consulte [Usando uma CDN para otimizar a personalização de MediaTailor anúncios e a entrega de conteúdo](#)

Tipo de manifesto de origem DASH

Se o servidor de origem produzir manifestos DASH de período único, abra a lista suspensa e escolha SINGLE_PERIOD. Por padrão, o MediaTailor lida com manifestos DASH de vários períodos. Para obter mais informações, consulte [the section called “Integrando uma fonte MPEG-DASH”](#).

Local de mpd DASH

(Opcional conforme necessário para o DASH) O local da descrição da apresentação de mídia (mpd). Escolha DESATIVADO para a seguinte situação:

- Você configura as regras de roteamento da CDN para acessar MediaTailor manifestos.
- Você usa relatórios do lado do cliente ou seu player suporta redirecionamentos HTTP persistentes.

Para obter mais informações sobre o recurso Location (Local), consulte [the section called “Recurso de localização”](#).

Transcodifique o nome do perfil

O nome que associa essa configuração a um perfil de transcodificação personalizado. Esse nome substitui os padrões de transcodificação dinâmica de. MediaTailor Preencha esse campo somente se você já tiver configurado perfis personalizados com a ajuda do AWS Support.

Passagem de marcadores de anúncios

Para HLS, ativa ou desativa a passagem de marcadores de anúncios. Quando a transmissão do marcador de anúncio está ativada, os marcadores de EXT-X-SPLICEPOINT-SCTE35 anúncio MediaTailor passam por EXT-X-CUE-IN e do manifesto de origem para o manifesto MediaTailor personalizado. EXT-X-CUE-OUT Nenhuma lógica é aplicada aos valores do marcador do anúncio; eles são passados do manifesto de origem para o manifesto personalizado no estado em que se encontram. Por exemplo, se EXT-X-CUE-OUT tiver um valor de 60 no manifesto de origem, mas nenhum anúncio for colocado, MediaTailor não alterará o valor para 0 no manifesto personalizado.

Visualizando detalhes da MediaTailor configuração

Além dos valores fornecidos quando a configuração foi criada, AWS Elemental MediaTailor exibe o nome da configuração, os endpoints de reprodução e o acesso relevante. URLs Para visualizar uma configuração, use o procedimento a seguir.

Como exibir uma configuração

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. Na página Configurations (Configurações), escolha o Configuration name (Nome da configuração) a ser exibida.

Editando MediaTailor as configurações

Você pode editar uma configuração para atualizar o servidor de origem e o mapeamento do servidor de decisão de anúncios (ADS) ou alterar a forma como AWS Elemental MediaTailor interage com uma rede de distribuição de conteúdo (CDN).

Para editar uma configuração

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. Na página Configurations (Configurações), escolha o nome da configuração que deseja editar.
3. Na página de detalhes da configuração, escolha Edit (Editar) e revise as definições de configuração conforme necessário. Não edite o nome da configuração. Para obter informações sobre atributos da configuração, consulte [Criando uma MediaTailor configuração de reprodução](#).
4. Escolha Salvar.

Excluindo configurações MediaTailor

Você pode excluir uma configuração para torná-la indisponível para reprodução com. AWS Elemental MediaTailor

Como excluir uma configuração

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.

2. Na página Configurations (Configurações), faça o seguinte:
 - Escolha o nome da configuração que você deseja excluir.
 - Na coluna Configuration name (Nome da configuração), escolha a opção ao lado do nome e Delete (Excluir).
3. Na caixa de confirmação Delete (Excluir), digite **Delete** e escolha Delete (Excluir).

Integrando uma fonte de conteúdo para inserção de MediaTailor anúncios

Este tópico aborda a integração de diferentes tipos de fontes de conteúdo de vídeo com o MediaTailor. O MediaTailor suporta os protocolos de streaming HLS e DASH para conteúdo ao vivo e sob demanda. O serviço pode realizar a inserção ou substituição de anúncios durante intervalos publicitários designados e tem requisitos específicos para a estrutura e formatação dos manifestos de vídeo de entrada para habilitar esses recursos. Os tópicos a seguir fornecem detalhes sobre os requisitos da fonte de entrada e as etapas para integrar o conteúdo do HLS e do DASH MediaTailor para permitir experiências publicitárias personalizadas.

Tópicos

- [Requisitos de fonte de entrada para inserção de MediaTailor anúncios](#)
- [Integrando uma fonte HLS](#)
- [Integrando uma fonte MPEG-DASH](#)
- [Protegendo as interações de AWS Elemental MediaTailor origem com o SigV4](#)

Requisitos de fonte de entrada para inserção de MediaTailor anúncios

Uma fonte de entrada deve atender aos seguintes requisitos para trabalhar com MediaTailor:

- Usar Apple HLS (HTTP Live Streaming) ou MPEG DASH (Dynamic Adaptive Streaming over HTTP)
- Usar streaming ao vivo ou VOD
- Estar acessível na internet pública e ter um endereço IP público
- Use portas HTTP padrão (porta 80) ou portas HTTPS (porta 443). MediaTailor não oferece suporte a portas personalizadas para comunicação com o servidor de origem.

- Contenha marcadores de anúncios em um dos formatos descritos no tutorial de [introdução à inserção de MediaTailor anúncios](#)

Integrando uma fonte HLS

AWS Elemental MediaTailor suporta .m3u8 manifestos HLS com um EXT-X-VERSION de 3 ou superior para transmissão ao vivo e vídeo sob demanda (VOD). Quando MediaTailor encontra uma quebra de anúncio, ele tenta inserir ou substituir o anúncio, com base no tipo de conteúdo. Se não houver anúncios suficientes para preencher a duração, durante o restante do intervalo publicitário, MediaTailor exibirá o fluxo de conteúdo subjacente ou a lista configurada. Para obter mais informações sobre o comportamento do anúncio HLS com base no tipo de conteúdo, consulte [Compreender o AWS Elemental MediaTailor comportamento de inserção de anúncios](#).

As seções a seguir fornecem mais informações sobre como MediaTailor lidar com manifestos HLS.

Tópicos

- [Marcadores de anúncios compatíveis com HLS](#)
- [Habilitando a passagem de marcadores de anúncios](#)
- [Manipulação de tags de manifesto HLS](#)
- [Exemplos de manifestos HLS](#)

Marcadores de anúncios compatíveis com HLS

AWS Elemental MediaTailor identifica os limites de disponibilidade de anúncios em um manifesto HLS analisando o manifesto de entrada em busca de marcadores de anúncios compatíveis. As seções a seguir descrevem quais marcadores o MediaTailor usa.

EXT-X-ASSET

A tag EXT-X-ASSET contém metadados usados pelo servidor de decisões de anúncios (ADS) para personalizar o conteúdo para o visualizador. Os parâmetros EXT-X-ASSET são pares de chave-valor separados por vírgula.

Para usar essa tag, é necessário atender aos seguintes requisitos:

- Você deve codificar em URL EXT-X-ASSET os valores no manifesto de origem. O exemplo a seguir mostra a tag EXT-X-ASSET com chaves e valores codificados em URL.

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

- Você deve incluir a `[asset.]` variável dinâmica e as chaves na configuração do MediaTailor ADS. O exemplo a seguir mostra uma configuração do MediaTailor ADS usando a `[asset.]` variável dinâmica e as chaves.

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

Exemplo de solicitação VAST

O exemplo a seguir mostra uma solicitação GET VAST para um ADS.

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

EXT-X-CUE-OUT and EXT-X-CUE-IN

Esse tipo de marcador de anúncio é o mais comum. Os exemplos a seguir mostram as opções para esses marcadores de deixas.

```
#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**

Com tags de marcadores de anúncios EXT-X-DATERANGE, use atributos SCTE35-OUT para especificar a programação da disponibilidade de anúncios.

Note

AWS Elemental MediaTailor ignora todos START-DATE os atributos fornecidos para marcadores de EXT-X-DATERANGE anúncios.

É possível especificar a disponibilidade de anúncios de uma das seguintes formas:

- A tag EXT-X-DATERANGE com especificações SCTE35-OUT e DURATION.

Exemplo

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

- Tags EXT-X-DATERANGE emparelhadas, a primeira com uma especificação SCTE35-OUT e a segunda com uma especificação SCTE35-IN.

Exemplo

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

- Uma combinação das opções anteriores. Você especifica uma tag EXT-X-DATERANGE com especificações DURATION SCTE35-OUT e seguidas por uma tag EXT-X-DATERANGE com uma especificação SCTE35-IN. Nesse caso, MediaTailor usa a configuração de sinalização mais antiga das duas especificações.

Exemplo

```
#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
```

PONTO DE EMENDA EXT-X- SCTE35

Anexe a tag do marcador de anúncios EXT-X-SPLICEPOINT-SCTE35 com uma carga SCTE-35 em binário codificado em base64. O binário codificado deve fornecer um `splice_info_section` SCTE-35 contendo o marcador de deixa de saída 0x34, para início da oportunidade de posicionamento do provedor, e o marcador de deixa de entrada 0x35, para término da oportunidade de posicionamento do provedor.

O exemplo a seguir mostra a especificação de ponto de emenda com cargas binárias codificadas em base64 que especificam marcadores de deixa de entrada e saída.

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

Habilitando a passagem de marcadores de anúncios

Por padrão, para o HLS, os manifestos MediaTailor personalizados não incluem os marcadores de anúncios SCTE-35 dos manifestos de origem. Quando a transmissão do marcador de anúncio está ativada, MediaTailor passa pelos seguintes marcadores de anúncio dos manifestos de origem para os manifestos personalizados:

- EXT-X-CUE-IN
- EXT-X-CUE-OUT
- PONTO DE EMENDA EXT-X- SCTE35

A passagem do marcador de anúncio é uma configuração opcional. Use a passagem de marcadores de anúncios se quiser que os marcadores de anúncios SCTE sejam incluídos no manifesto personalizado. MediaTailor Os casos de uso comuns incluem o seguinte:

- Substituição de conteúdo - Execute a substituição ou restrição de conteúdo.
- Rastreamento de anúncios: acione as informações de rastreamento de anúncios com base na presença ou ausência de um ou mais marcadores de anúncios.
- Configurações do player - Ative a funcionalidade de depuração ou cronômetro de contagem regressiva na interface do player, com base na presença ou ausência de marcadores de anúncios.

Note

MediaTailor não altera os valores desses marcadores. Por exemplo, se EXT-X-CUE-OUT tiver um valor de 60 no manifesto de origem, mas nenhum anúncio for colocado, MediaTailor não alterará o valor para 0 no manifesto personalizado.

Ativar a passagem de marcadores de anúncios

Você pode ativar a passagem do marcador de anúncio usando o Console de gerenciamento da AWS ou o AWS Command Line Interface (AWS CLI).

Para ativar a passagem de marcadores de anúncios usando o console

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. Selecione Nova configuração ou Editar configuração.
3. Na seção Configurações avançadas, selecione Ativar no menu suspenso.

Para ativar a passagem de marcadores de anúncios usando o AWS Command Line Interface (AWS CLI)

Use o comando [put-playback-configuration](#).

Manipulação de tags de manifesto HLS

Esta seção descreve como AWS Elemental MediaTailor gerencia as tags no manifesto de saída personalizado.

EXT-X-CUE etiquetas

MediaTailor substitui as EXT-X-CUE-IN tags EXT-X-CUE-OUTEXT-X-CUE-OUT-CONT, e no manifesto de entrada por EXT-X-DISCONTINUITY tags no manifesto de saída. As tags DISCONTINUITY marcam os seguintes limites:

- Onde o conteúdo principal muda para um anúncio
- Onde um anúncio muda para outro anúncio
- Onde um anúncio muda de volta para o conteúdo principal

Tags do EXT-X-DATERANGE

MediaTailor passa pelas EXT-X-DATERANGE tags do manifesto de entrada para o manifesto de saída. MediaTailor também insere EXT-X-DISCONTINUITY tags que correspondem às DATERANGE tags. As tags DISCONTINUITY marcam os seguintes limites:

- Onde o conteúdo principal muda para um anúncio
- Onde um anúncio muda para outro anúncio
- Onde um anúncio muda de volta para o conteúdo principal

Tags do EXT-X-KEY

MediaTailor passa pelas EXT-X-KEY tags do manifesto de entrada. Essas tags indicam que o conteúdo principal é criptografado. Como os anúncios não são criptografados, o MediaTailor insere EXT-X-KEY:METHOD=NONE no início de uma disponibilidade de anúncio. Quando a reprodução retornar ao conteúdo principal, MediaTailor reative a criptografia inserindo a EXT-X-KEY tag com o METHOD valor definido como o tipo de criptografia.

Tags não reconhecidas

MediaTailor passa por todas as tags desconhecidas e personalizadas do manifesto de entrada para o manifesto de saída.

Exemplos de manifestos HLS

As seções a seguir fornecem exemplos de manifestos de origem HLS e manifestos personalizados. Compreender esses exemplos pode ajudá-lo a configurar e solucionar problemas em seus MediaTailor fluxos de trabalho.

Para obter informações sobre como os parâmetros de consulta são aplicados aos manifestos e segmentos do HLS, consulte [MediaTailor Inicialização de sessão implícita do HLS](#)

Entendendo os tipos de playlists do HLS

O HTTP Live Streaming (HLS) usa dois tipos principais de playlists:

Lista de reprodução multivariante

Uma playlist multivariante é o arquivo de índice de nível superior que lista todas as representações disponíveis do conteúdo. Ele contém referências a playlists de mídia, mas não contém nenhum segmento de mídia em si. Essa playlist permite que os jogadores selecionem a representação mais adequada com base nas condições da rede, nos recursos do dispositivo ou nas preferências do usuário.

Esse tipo de playlist também é conhecido por vários outros nomes em vários contextos, incluindo playlist master, manifesto principal, playlist primária, playlist principal, arquivo de índice ou master M3U8.

Em MediaTailor fluxos de trabalho, a playlist multivariante é o ponto de entrada para solicitações de reprodução e é onde a personalização de anúncios começa.

Playlist de mídia

Uma playlist de mídia contém as informações reais do segmento de mídia para uma representação específica (nível de qualidade) do conteúdo. Ele inclui informações de tempo URLs, segmentos e outros metadados necessários para a reprodução de uma única representação.

Esse tipo de playlist também é conhecido como playlist de mídia, manifesto secundário, chunklist, mídia M3U8 ou playlist de renderização.

Nos MediaTailor fluxos de trabalho, as playlists de mídia são personalizadas para incluir segmentos de conteúdo e segmentos de anúncios na sequência correta.

Para obter informações mais detalhadas sobre os tipos de playlist do HLS, consulte [Tipos de playlist HLS](#).

Exemplos de manifesto de origem do HLS

O exemplo a seguir mostra uma playlist multivariante HLS AWS Elemental MediaTailor recebida pelo HLS da origem do conteúdo.

```
#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_
```

Neste exemplo de playlist multivariante:

- As `#EXT-X-STREAM-INF` tags definem diferentes rendições com diferentes resoluções e taxas de bits
- Cada representação faz referência a uma playlist de mídia (como) `index_1.m3u8`
- A `#EXT-X-MEDIA` tag define uma faixa de legenda

O exemplo a seguir mostra uma playlist de mídia HLS AWS Elemental MediaTailor recebida pelo HLS da origem do conteúdo. Esse exemplo usa tags `EXT-X-CUE-OUT` e `EXT-X-CUE-IN` para descrever oportunidades de disponibilidade de anúncios.

```
#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

```

Neste exemplo de playlist de mídia:

- As #EXTINF tags especificam a duração de cada segmento
- A #EXT-X-CUE-OUT tag marca o início de um intervalo publicitário
- As #EXT-X-CUE-OUT-CONT tags fornecem informações sobre o intervalo publicitário em andamento
- A #EXT-X-CUE-IN tag marca o final do intervalo publicitário

Exemplos de manifestos personalizados do HLS

O exemplo a seguir mostra uma playlist multivariante HLS personalizada. AWS Elemental MediaTailor

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

Observe como MediaTailor modificou a playlist de mídia URLs para incluir informações específicas da sessão que permitem a inserção personalizada de anúncios.

O exemplo a seguir mostra uma playlist de mídia AWS Elemental MediaTailor personalizada.

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:8779957
#EXT-X-DISCONTINUITY-SEQUENCE:0
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779957.ts?m=1566416212
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779958.ts?m=1566416212
#EXTINF:5.372,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779959.ts?m=1566416212
#EXT-X-DISCONTINUITY
```

```

#EXTINF:3.066667,
../.../segment/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779960
#EXTINF:3.0,
../.../segment/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779961
#EXTINF:3.0,
../.../segment/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779962
#EXTINF:3.0,
../.../segment/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779963
#EXTINF:2.966667,
../.../segment/43f3e412052f2808dd84ea1da90e92e914edddde/external-canary-hls/
ee1696a8-4f7f-4c4c-99de-9821131847e8/0/8779964
#EXT-X-DISCONTINUITY
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779963.ts?m=1566416212
#EXTINF:1.368,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779964.ts?m=1566416212
#EXTINF:4.638,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779965.ts?m=1566416212
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779966.ts?m=1566416212
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779967.ts?m=1566416212
#EXTINF:6.006,
https://777788889999.mediapackage.us-west-2.amazonaws.com/out/v1/
e309ffd02ba8498d864dcaacff7a5ad9/index_1_8779968.ts?m=1566416212

```

Neste exemplo de playlist de mídia personalizada:

- MediaTailor inseriu segmentos de anúncios entre os segmentos de conteúdo
- As #EXT-X-DISCONTINUITY tags marcam as transições entre conteúdo e anúncios
- Os segmentos de conteúdo são servidos a partir do servidor de origem (MediaPackage neste exemplo)

- Os segmentos de anúncios são servidos a partir MediaTailor do armazenamento de segmentos de anúncios

Principais diferenças em manifestos personalizados

Quando MediaTailor personaliza os manifestos do HLS, ele faz várias mudanças importantes:

Mudanças multivariantes na playlist

- URLs As playlists variantes são reescritas para apontar para MediaTailor -managed, incluindo URLs informações da sessão
- A ordem das tags pode ser reorganizada para uma reprodução ideal

Alterações na playlist variante

- Os marcadores de anúncios (EXT-X-CUE-OUT,EXT-X-CUE-IN) são substituídos por segmentos de anúncios reais
- Marcadores de descontinuidade (EXT-X-DISCONTINUITY) são adicionados nos limites content/ad
- URLs Os segmentos de conteúdo são reescritos para apontar para a origem ou CDN
- Segmentos de anúncios URLs são adicionados ao armazenamento de segmentos MediaTailor de anúncios do point to

Compreender essas mudanças pode ajudá-lo a solucionar problemas em seus MediaTailor fluxos de trabalho e garantir a configuração adequada do CDN e do player.

Tópicos relacionados

Para obter mais informações sobre manifestos do HLS e MediaTailor, consulte os seguintes tópicos:

- [Tipos de playlist HLS](#)- Explicação detalhada dos tipos de playlists HLS
- [Usando uma CDN para otimizar a personalização de MediaTailor anúncios e a entrega de conteúdo](#)- Informações sobre o uso de uma CDN com MediaTailor
- [Como funciona a inserção de MediaTailor anúncios](#)- Visão geral de como a inserção de MediaTailor anúncios funciona

Integrando uma fonte MPEG-DASH

AWS Elemental MediaTailor suporta manifestos .mpd ao vivo e de vídeo sob demanda (VOD) que seguem as diretrizes do perfil dinâmico do DASH. MediaTailor aceita entradas de manifesto compatíveis com DASH de vários períodos e de período único e fornece saídas de manifesto compatíveis com DASH de vários períodos.

Os manifestos de entrada devem ter o seguinte:

- Streamings de eventos SCTE-35 com configurações de informações de emenda para `splice insert` ou `time signal`. As configurações podem ser fornecidas em XML limpo ou em binário codificado em base64.
- `Segment templates` com `segment timelines`.

Para manifestos publicados, MediaTailor exige que as atualizações do servidor de origem deixem o seguinte inalterado:

- Horários de início do período, especificados no atributo `start`.
- Valores de `presentationTimeOffset` nos modelos de segmento das representações do período.

Como prática recomendada, forneça aos anúncios as mesmas `Representation` configurações `AdaptationSet` e os mesmos prazos do fluxo de conteúdo. AWS Elemental MediaTailor usa essas configurações para transcodificar os anúncios de acordo com o fluxo de conteúdo, para uma alternância suave entre os dois.

As seções a seguir fornecem mais informações sobre como MediaTailor lidar com anúncios em manifestos do DASH.

Tópicos

- [Marcadores de anúncios DASH](#)
- [DASH e duração do Avail](#)
- [Numeração de segmentos do manifesto DASH](#)
- [Exemplos de DASH MPD](#)
- [Recurso de localização do DASH](#)

Marcadores de anúncios DASH

MediaTailor

AWS Elemental MediaTailor usa marcadores de saída SCTE-35 para identificar anúncios no manifesto DASH usando a seguinte lógica:

- DASH de vários períodos: MediaTailor insere anúncios para o primeiro Event em cada um Period que contém marcadores SpliceInsert ou TimeSignal marcadores destacados. MediaTailor ignora Event marcadores adicionais no Period
- DASH de período único: MediaTailor insere anúncios Period que contém Event marcadores SpliceInsert ou TimeSignal marcadores destacados.

Por padrão, AWS Elemental MediaTailor gerencia os manifestos do DASH como manifestos de vários períodos. Você pode alterar sua configuração para lidar com manifestos DASH de um único período no servidor de origem. Para mais informações, consulte [the section called “Criar uma configuração”](#).

As seções a seguir fornecem detalhes adicionais sobre o manuseio de marcadores de anúncios DASH e fornecem manifestos decorados desde a origem.

Requisitos de XML do manifesto de origem do DASH

Os marcadores de anúncios nos manifestos do DASH da origem devem ser formatados adequadamente MediaTailor para identificar quebras de anúncios. Os tópicos a seguir descrevem esses requisitos de formatação em XML claro.

SpliceInsert em XML claro

SpliceInsertos marcadores de anúncios em XML claro devem conter o seguinte:

- EventStream deve ter o seguinte atributo: `schemeIdUri=urn:scte:scte35:2013:xml`
- Event deve aguentar `scte35:SpliceInfoSection`
- `scte35:SpliceInfoSection` deve aguentar `scte35:SpliceInsert`
- `scte35:SpliceInsert` deve ter o seguinte atributo: `outOfNetworkIndicator="true"`

Exemplo **SpliceInsert** em XML

No exemplo a seguir, os marcadores SCTE obrigatórios estão em negrito.

```
<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>
```

TimeSignal em XML claro

TimeSignal os marcadores de anúncios em XML claro devem conter o seguinte:

- EventStream deve ter o seguinte atributo: `schemeIdUri=urn:scte:scte35:2013:xml`
- Event deve aguentar `scte35:SpliceInfoSection`
- `scte35:SpliceInfoSection` deve aguentar `scte35:TimeSignal`
- `scte35:SpliceInfoSection` também deve segurar `scte35:SegmentationDescriptor`
- `scte35:SegmentationDescriptor` deve ter o seguinte atributo, em que o valor é válido [Números de recorte](#): `segmentationTypeId="xx"`
- `scte35:SegmentationDescriptor` deve aguentar `scte35:SegmentationUpid`

Números de recorte

A seguir estão os números de sinalização suportados para o `segmentationTypeId`

Mensagem de segmentação	Valor do <code>segmentationTypeId</code>	Valor hexadecimal
Fim do anúncio do distribuidor	51	0x51

Mensagem de segmentação	Valor do segmentationTypeId	Valor hexadecimal
Início do anúncio do distribuidor	50	0x32
Fim da oportunidade de colocação de distribuidores	55	0x37
Início da oportunidade de colocação de distribuidor	54	0x36
Intervalo final	35	0x23
Fim do anúncio do provedor	49	0x31
Início do anúncio do provedor	48	0x30
Fim da oportunidade de colocação de sobreposição de fornecedores	57	0x39
Início da oportunidade de colocação de sobreposição de fornecedores	56	0x38
Fim da oportunidade de colocação de fornecedores	53	0x35
Início da oportunidade de colocação de fornecedores	52	0x34
Comece o intervalo	34	0x22

Exemplo **TimeSignal** em XML

No exemplo a seguir, os marcadores SCTE obrigatórios estão em negrito.

```
<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>
.
.
.
</Period>

```

Requisitos binários codificados em base64 do manifesto de origem do DASH

Os marcadores de anúncios nos manifestos do DASH da origem devem ser formatados adequadamente MediaTailor para identificar quebras de anúncios. Os tópicos a seguir descrevem esses requisitos de formatação em binário codificado em base64.

Ambos `TimeSignal` e os marcadores de `SpliceInsert` anúncios em manifestos codificados em base64 devem conter o seguinte:

- `EventStream` deve ter o seguinte atributo: `urn:scte:scte35:2014:xml+bin`
- `Event` deve aguentar `scte35:Signal`
- `scte35:Signal` deve conter um `scte35:Binary` binário codificado em base64.

O binário decodificado deve fornecer a `splice_info_section` com as mesmas informações necessárias para marcadores de anúncios em XML claros.

- O tipo de comando deve ser `splice_insert()` ou `time_signal()`
- As configurações adicionais devem estar em conformidade com as descritas em [TimeSignal em XML claro](#) [SpliceInsert em XML claro](#) e.

O binário codificado deve fornecer um `splice_info_section` com o mesmo conjunto de informações que o XML limpo forneceria em um elemento `scte35:SpliceInfoSection`. O tipo de comando deve ser `splice_insert()` ou `time_signal()`, e as configurações adicionais devem estar em conformidade com as descritas anteriormente para a entrega de XML limpo.

O exemplo a seguir mostra essa opção, com os marcadores necessários em negrito.

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

        <scte35:Binary>QW5vdGhlciB0ZXN0IHN0cmVuZyBmb3IgaW5jb2RpbmcgdG8gQmFzZTY0IGVuY29kZWQgYmluYXJ5Lg=
scte35:Binary>
        </scte35:Signal>
      </Event>
      .
      .
      .
    </Period>
```

O seguinte é o binário decodificado para o primeiro evento listado no exemplo anterior. A configuração `splice_command_type` é 5, o que indica `splice_insert`.

```
{
  "table_id": 252,
  "section_syntax_indicator": false,
  "private_indicator": false,
  "section_length": 33,
  "protocol_version": 0,
  "encrypted_packet": false,
  "encryption_algorithm": 0,
  "pts_adjustment": 0,
  "cw_index": 0,
  "tier": "0xFFF",
  "splice_command_length": 16,
```

```

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

```

DASH e duração do Avail

Durante a reprodução, quando AWS Elemental MediaTailor encontra um anúncio disponível, ele substitui parte ou toda a disponibilidade por anúncios. MediaTailor inicia a substituição do anúncio no início da disponibilidade do anúncio e inclui anúncios da seguinte forma:

- Se a disponibilidade do anúncio especificar uma duração, MediaTailor inclua quantos anúncios couberem dentro do limite de duração, sem substituir o conteúdo a seguir.
- Se nenhuma duração for fornecida, MediaTailor inclua anúncios até o final da disponibilidade do anúncio. Para manifestos de vários períodos, este é o término do período. Para manifestos de período único, esse é o fim do evento. MediaTailor não exibe anúncios após o final da disponibilidade do anúncio e, quando chega ao fim, trunca o anúncio atual em vez de sobrescrever o conteúdo a seguir.

Como AWS Elemental MediaTailor parece a duração da disponibilidade do anúncio

AWS Elemental MediaTailor pesquisa uma configuração de duração na seguinte ordem:

1. Event duration
2. Para inserção por emenda, scte35:BreakDuration duration
3. Para o sinal de tempo, scte35:SegmentationDescriptor segmentationDuration

Se AWS Elemental MediaTailor não encontrar nenhuma dessas configurações, ele gerencia a inclusão de anúncios sem uma duração.

O exemplo a seguir mostra um recurso Event que tem uma 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>
```

O exemplo a seguir mostra a disponibilidade de anúncios sem duração especificada. O Event não tem duration, e o elemento `scte35:SpliceInsert` não contém um elemento filho `scte35:BreakDuration`.

```
<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>
```

Numeração de segmentos do manifesto DASH

MediaTailor suporta segmentos de mídia `<SegmentTemplate>` que são definidos usando `<SegmentTimeline>` e o `media` atributo. Você pode especificar a lista de segmentos de mídia no atributo `media` usando o identificador `$Number$` ou o identificador `$Time$`.

O exemplo a seguir mostra um `SegmentTemplate` com um atributo `media` que usa o identificador `$Number$`.

```
<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>
```

O exemplo a seguir mostra um `SegmentTemplate` com um atributo `media` que usa o identificador `$Time$`.


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

Exemplos de DASH MPD

As seções a seguir fornecem exemplos de origem MPDs e personalização MPDs do DASH. Compreender esses exemplos pode ajudá-lo a configurar e solucionar problemas em seus MediaTailor fluxos de trabalho.

Para obter informações sobre como os parâmetros de consulta são aplicados aos manifestos e segmentos do DASH, consulte [MediaTailor Inicialização de sessão implícita do DASH](#)

Entendendo a estrutura do DASH MPD

O Dynamic Adaptive Streaming over HTTP (DASH) usa um manifesto Media Presentation Description (MPD) para fornecer conteúdo de streaming. O MPD é um documento XML que descreve a estrutura e a disponibilidade do conteúdo de mídia.

MPD (Descrição da apresentação de mídia)

O MPD é o arquivo de manifesto principal no streaming do DASH que descreve a estrutura e a disponibilidade do conteúdo de mídia. Ele contém informações sobre períodos, conjuntos de adaptação, representações e segmentos que compõem o conteúdo de streaming.

Esse tipo de manifesto também é conhecido por vários outros nomes em vários contextos, incluindo manifesto DASH, DASH MPD, manifesto mestre (quando comparado ao HLS) ou manifesto de apresentação.

Em MediaTailor fluxos de trabalho, o MPD é o ponto de entrada para solicitações de reprodução e é onde a personalização de anúncios começa.

Período

Um período é uma seção temporal de uma apresentação do DASH. Cada período contém um ou mais conjuntos de adaptação e representa um período de tempo de mídia. Em fluxos de trabalho

de inserção de anúncios, períodos separados são normalmente usados para delimitar entre conteúdo e anúncios.

Nos MediaTailor fluxos de trabalho, os períodos são usados para separar o conteúdo principal do conteúdo do anúncio, com cada anúncio normalmente representado por seu próprio período.

AdaptationSet

E AdaptationSet agrupa um conjunto de versões codificadas intercambiáveis de um ou vários componentes de conteúdo de mídia. Por exemplo, um AdaptationSet pode conter vários níveis de qualidade de vídeo, enquanto outro pode conter várias opções de idioma de áudio.

Nos MediaTailor fluxos de trabalho, AdaptationSets são preservados durante a inserção do anúncio para manter os tipos de mídia consistentes entre o conteúdo e os anúncios.

Representação

Uma representação é uma versão codificada específica do conteúdo de mídia em um AdaptationSet. Cada representação normalmente difere em taxa de bits, resolução ou outros parâmetros de codificação, permitindo que os clientes selecionem a versão mais apropriada com base nas condições da rede e nos recursos do dispositivo.

Em MediaTailor fluxos de trabalho, as Representações em Períodos de Anúncios são combinadas da forma mais próxima possível com as Representações em Períodos de Conteúdo para garantir uma experiência de visualização tranquila.

Para obter informações mais detalhadas sobre os tipos de manifesto DASH, consulte [Tipos de manifesto DASH](#).

Para obter informações sobre a estrutura e configuração do manifesto do DASH em AWS Elemental MediaPackage, consulte a seção Guia do MediaPackage usuário sobre a visão geral do DASH.

Exemplos de DASH MPD ao vivo

Esta seção fornece exemplos de DASH MPDs ao vivo. Cada exemplo lista um MPD como recebido do servidor de origem e depois MediaTailor personalizou o MPD com anúncios.

Exemplo de inserção de emenda DASH MPD

Exemplo de MPD de origem DASH para inserção de emenda

O exemplo a seguir de um MPD mostra um anúncio disponível em um manifesto recebido pelo DASH da origem do conteúdo. Este exemplo usa o `SpliceInsert` marcador para indicar a disponibilidade de um anúncio.

```
<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:SegmentationDescriptor segmentationEventId="99"
segmentationEventCancelIndicator="false" segmentationDuration="9450000">
          <scte35:DeliveryRestrictions webDeliveryAllowedFlag="true"
noRegionalBlackoutFlag="true" archiveAllowedFlag="true" deviceRestrictions="3"/>
          <scte35:SegmentationUpid segmentationUpidType="8" segmentationUpidLength="0"/>
        </scte35:SegmentationDescriptor>
        <scte35:SegmentationTypeID segmentationType="52"/>
        <scte35:SegmentNum segmentNum="1"/>
        <scte35:SegmentsExpected segmentsExpected="1"/>
      </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>

```

Neste exemplo de origem do MPD:

- O <EventStream> elemento contém marcadores SCTE-35 que indicam anúncios disponíveis
- O <scte35:SpliceInsert> elemento fornece detalhes sobre a disponibilidade do anúncio
- O <scte35:BreakDuration> elemento especifica a duração do intervalo publicitário
- Os <AdaptationSet> elementos definem os fluxos de vídeo e áudio disponíveis

Exemplo de MPD personalizado do DASH para inserção de emenda

AWS Elemental MediaTailor personaliza o anúncio com as especificações de publicidade. As personalizações refletem os dados do espectador recebidos do player e as campanhas publicitárias atualmente em andamento.

O exemplo a seguir mostra a disponibilidade de um anúncio depois de AWS Elemental MediaTailor personalizá-lo.

```

<Period id="178443_1" start="PT96H15M30.25S">
  <BaseURL>http://111122223333.cloudfront.net/nbc_fallback_2/</BaseURL>

```

```

<AdaptationSet bitstreamSwitching="false" frameRate="30/1" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
  <SegmentTemplate startNumber="1" timescale="90000"/>
  <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>
</AdaptationSet>
<AdaptationSet mimeType="audio/mp4" segmentAlignment="0" lang="eng">
  <Representation id="8" bandwidth="128000" audioSamplingRate="48000"
codecs="mp4a.40.2">
    <SegmentTemplate initialization="nbc_fallback_ad_2_audio_2init.mp4"
media="nbc_fallback_ad_2_audio_2_${Number%09d$.mp4" startNumber="1" timescale="90000">
      <SegmentTimeline>
        <S d="180000" r="13" t="0"/>
        <S d="176940" t="2520000"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </Representation>
</AdaptationSet>
</Period>

```

Neste exemplo personalizado de MPD:

- MediaTailor criou um novo período para o conteúdo do anúncio
- O <BaseURL> elemento aponta para o local do conteúdo do anúncio

- Os <AdaptationSet> elementos mantêm uma estrutura semelhante ao conteúdo
- Os <Representation> elementos fornecem diferentes níveis de qualidade para o conteúdo do anúncio

Exemplo de sinal de horário DASH MPD

Exemplo de MPD de origem DASH para sinal de tempo

O exemplo a seguir de um MPD mostra um anúncio disponível em um manifesto recebido pelo DASH da origem do conteúdo. Este exemplo usa o `TimeSignal` marcador para indicar a disponibilidade de um anúncio.

```
<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:TimeSignal>
          <scte35:SpliceTime ptsTime="7835775000"/>
        </scte35:TimeSignal>
        <scte35:SegmentationDescriptor segmentationEventId="99"
segmentationEventCancelIndicator="false" segmentationDuration="9450000">
          <scte35:DeliveryRestrictions webDeliveryAllowedFlag="true"
noRegionalBlackoutFlag="true" archiveAllowedFlag="true" deviceRestrictions="3"/>
          <scte35:SegmentationUpid segmentationUpidType="8" segmentationUpidLength="0"/>
        </scte35:SegmentationDescriptor>
        <scte35:SegmentationTypeID segmentationType="52"/>
        <scte35:SegmentNum segmentNum="1"/>
        <scte35:SegmentsExpected segmentsExpected="1"/>
      </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>
</Period>

```

Neste exemplo de origem do MPD:

- O `<scte35:TimeSignal>` elemento é usado em vez de `<scte35:SpliceInsert>`
- `<scte35:SegmentationDescriptor>` Fornece informações adicionais sobre a disponibilidade do anúncio

Exemplo de MPD personalizado do DASH para sinal de tempo

AWS Elemental MediaTailor personaliza o anúncio com as especificações de publicidade. As personalizações refletem os dados do espectador recebidos do player e as campanhas publicitárias atualmente em andamento.

O exemplo a seguir mostra a disponibilidade de um anúncio depois de AWS Elemental MediaTailor personalizá-lo.

```

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

```

O MPD personalizado para sinal de tempo é semelhante ao da inserção de emenda, com a MediaTailor criação de um novo período para o conteúdo do anúncio.

Exemplos de VOD DASH MPD

Esta seção fornece exemplos de vídeo sob demanda (VOD) MPDs DASH. Cada exemplo lista um MPD como recebido do servidor de origem e depois MediaTailor personalizou o MPD com anúncios.

O VOD DASH MPDs segue a mesma estrutura do live MPDs, mas normalmente tem um `type="static"` atributo no elemento MPD e pode conter vários períodos para diferentes segmentos de conteúdo.

Para exemplos de VOD DASH MPDs, consulte a MediaTailor documentação em: [Marcadores de anúncios DASH](#)

Principais diferenças na personalização MPDs

Quando MediaTailor personaliza o DASH MPDs, ele faz várias mudanças importantes:

Tratamento do período

- Novos períodos são criados para o conteúdo do anúncio
- Os horários de início do período são ajustados para manter a continuidade do cronograma
- EventStream elementos com marcadores SCTE-35 são processados e removidos

AdaptationSet e tratamento de representações

- AdaptationSets no anúncio Os períodos são criados para corresponder ao conteúdo AdaptationSets
- As representações são criadas para diferentes níveis de qualidade do conteúdo do anúncio
- SegmentTemplate os elementos são atualizados para apontar para o conteúdo do anúncio

Compreender essas mudanças pode ajudá-lo a solucionar problemas em seus MediaTailor fluxos de trabalho e garantir a configuração adequada do CDN e do player.

Tópicos relacionados

Para obter mais informações sobre o DASH MPDs e MediaTailor, consulte os seguintes tópicos:

- [Tipos de manifesto DASH](#)- Explicação detalhada dos tipos de manifesto DASH

- [Usando uma CDN para otimizar a personalização de MediaTailor anúncios e a entrega de conteúdo](#)- Informações sobre o uso de uma CDN com MediaTailor
- [Como funciona a inserção de MediaTailor anúncios](#)- Visão geral de como a inserção de MediaTailor anúncios funciona
- Para obter informações abrangentes sobre a estrutura e a MediaPackage configuração do manifesto do DASH, consulte a seção Guia do MediaPackage usuário sobre a visão geral do DASH.

Recurso de localização do DASH

Esta seção fornece informações sobre o recurso de local de DASH, habilitado por padrão em AWS Elemental MediaTailor. Leia esta seção se você criar regras de roteamento da rede de distribuição de conteúdo (CDN) para acessar MediaTailor manifestos. Também leia esta seção caso você use relatórios no lado do servidor com players que não ofereçam suporte a redirecionamentos HTTP recorrentes.

O que é o recurso de local?

O recurso de local permite que players que não ofereçam suporte a redirecionamentos HTTP recorrentes forneçam um comportamento recorrente nas solicitações de atualização do manifesto.

AWS Elemental MediaTailor usa inicialização sem sessão e exige um comportamento persistente de redirecionamento HTTP de seus jogadores. Com os relatórios do lado do servidor, quando o jogador solicita uma atualização do manifesto para MediaTailor, o serviço emite um redirecionamento 302 temporário, para direcionar o jogador a um endpoint para o manifesto personalizado. MediaTailor inclui um ID de sessão na resposta, como parâmetro de consulta. A intenção é para o player seguir o URL em toda a sessão, mas os players que não oferecem suporte a redirecionamentos HTTP recorrentes ignoram o redirecionamento e retornam ao URL original. Quando um player retorna ao URL original, para cada nova solicitação MediaTailor cria uma nova sessão em vez de permanecer com a sessão original. Isso pode corromper o manifesto.

A especificação DASH fornece uma solução para esse problema no recurso de localização, que é ativado por padrão nas AWS Elemental MediaTailor configurações. Quando esse recurso está ativado, MediaTailor coloca o URL absoluto na <Location> tag do manifesto. Os players que não oferecem suporte a redirecionamentos HTTP recorrentes podem usar o URL fornecido em <Location> para solicitar atualizações no manifesto.

Preciso desabilitar o recurso de local na minha configuração?

O recurso de local substitui todas as regras de roteamento CDN configuradas para acessar manifestos do AWS Elemental MediaTailor e, assim, talvez seja necessário desabilitá-lo. O recurso de local não afeta o armazenamento em cache CDN de conteúdo ou segmentos de anúncios.

Encontre a situação na lista a seguir para determinar se você precisa desabilitar o recurso de local para a configuração e como processar isso:

- Caso você não tenha regras de roteamento CDN configuradas para acessar manifestos do AWS Elemental MediaTailor, deixe a configuração do local habilitada.
- Do contrário, use as seguintes regras:
 - Caso você não use relatórios no lado do servidor ou todos os players ofereçam suporte a redirecionamentos HTTP recorrentes, desabilite o recurso de local. Para obter informações sobre como fazer isso no console, consulte [the section called “Criar uma configuração”](#).
 - Caso contrário, entre em contato com o [AWS Support](#).

Preciso usar o recurso de local?

Você precisa usar o recurso de localização para jogadores que não suportam redirecionamentos HTTP fixos. Use o URL fornecido na tag <Location> em todas as solicitações de atualização de manifesto.

Exemplo

Exemplo URLs e exemplo de <Location> tag.

- Example Exemplo: URL da solicitação inicial

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

- Example Exemplo: resposta 302 redirecionada

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

- Example Exemplo: tag de local em um manifesto

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

Protegendo as interações de AWS Elemental MediaTailor origem com o SigV4

O Signature Version 4 (SigV4) é um protocolo de assinatura usado para autenticar MediaTailor solicitações em origens suportadas via HTTPS. MediaTailor só suporta comunicação HTTPS e não permite conexões HTTP. Com a assinatura SigV4, MediaTailor inclui um cabeçalho de autorização assinado na solicitação de origem HTTPS para MediaTailor Channel Assembly, Amazon S3 AWS Elemental MediaPackage e versão 2.

Você pode usar o SigV4 em sua origem para garantir que as solicitações de manifesto sejam atendidas somente se forem de MediaTailor e contiverem um cabeçalho de autorização assinado. Dessa forma, configurações de MediaTailor reprodução não autorizadas são impedidas de acessar seu conteúdo original. Se o cabeçalho de autorização assinado for válido, sua origem atenderá à solicitação. Se não for válido, a solicitação falhará.

As seções a seguir descrevem os requisitos para usar a assinatura MediaTailor SigV4 nas origens suportadas.

MediaTailor Requisitos de montagem do canal

Se você usa o SigV4 para proteger sua origem de montagem de MediaTailor canais, os seguintes requisitos devem ser atendidos MediaTailor para acessar o manifesto:

- O URL base de origem em sua MediaTailor configuração deve ser um canal de montagem de canais no seguinte formato: `channel-assembly.mediatailor.region.amazonaws.com`
- Sua origem deve estar configurada para usar HTTPS. MediaTailor só suporta comunicação HTTPS e não permite conexões HTTP. Se o HTTPS não estiver habilitado na origem, não MediaTailor assinará a solicitação.
- Seu canal deve ter uma política de acesso de origem que inclua o seguinte:
 - Acesso principal MediaTailor para acessar seu canal. Conceda acesso a `mediataylor.amazonaws.com`.

- Mediataylor de permissões do IAM: GetManifest para ler todas as playlists multivariantes referenciadas pela configuração. MediaTailor

Para obter informações sobre como definir uma política no canal, consulte [Crie um canal usando o MediaTailor console](#).

Example política de acesso de origem para Channel Assembly, com escopo na conta MediaTailor de configuração

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

Example política de acesso de origem para Channel Assembly, com escopo de acordo com a configuração de MediaTailor reprodução

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

Requisitos do Amazon S3

Se você usa o SigV4 para proteger sua origem do Amazon S3, os seguintes requisitos devem ser atendidos MediaTailor para acessar o manifesto:

- O URL base de origem em sua MediaTailor configuração deve ser um bucket do S3 no seguinte formato: `s3.region.amazonaws.com`

- Sua origem deve estar configurada para usar HTTPS. MediaTailor só suporta comunicação HTTPS e não permite conexões HTTP. Se o HTTPS não estiver habilitado na origem, não MediaTailor assinará a solicitação.

- Seu canal deve ter uma política de acesso de origem que inclua o seguinte:

- Acesso principal MediaTailor para acessar seu bucket. Conceda acesso a `mediatailor.amazonaws.com`.

Para obter informações sobre como configurar o acesso no IAM, consulte [Gerenciamento de acesso](#) no Guia do usuário do AWS Identity and Access Management.

- Permissões do IAM `s3:GetObject` para ler todos os manifestos de nível superior referenciados pela configuração. MediaTailor

Para obter informações gerais sobre o SigV4 para o Amazon S3, consulte [o tópico Solicitações de autenticação \(AWS Signature versão 4\)](#) na referência da API do Amazon S3.

Exemplo política de acesso de origem para o Amazon S3, com escopo definido para a conta MediaTailor

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

Exemplo política de acesso de origem para o Amazon S3, com escopo de acordo com a configuração de reprodução MediaTailor

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

```
}
```

MediaPackage requisitos

Se você usa o SigV4 para proteger sua origem MediaPackage v2, os seguintes requisitos devem ser atendidos MediaTailor para acessar o manifesto:

- O URL base de origem em sua MediaTailor configuração deve ser um endpoint MediaPackage v2 no seguinte formato: `mediapackagev2.region.amazonaws.com`
- Sua origem deve estar configurada para usar HTTPS. MediaTailor só suporta comunicação HTTPS e não permite conexões HTTP. Se o HTTPS não estiver habilitado na origem, não MediaTailor assinará a solicitação.
- Seu canal deve ter uma política de acesso de origem que inclua o seguinte:
 - Acesso principal para MediaTailor acessar seu endpoint. Conceda acesso a `mediatailor.amazonaws.com`.
 - Permissões do IAM `mediapackagev2:GetObject` para ler todas as playlists multivariantes referenciadas pela configuração. MediaTailor

Para obter informações gerais sobre o SigV4 para MediaPackage v2, consulte o tópico [Solicitações de autenticação \(AWS Signature versão 4\)](#) na referência da MediaPackage API v2.

Exemplo política de acesso de origem para MediaPackage v2, com escopo definido para a conta MediaTailor

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

Exemplo política de acesso de origem para MediaPackage v2, com escopo de acordo com a configuração de reprodução MediaTailor

```
{
```

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

Integrar AWS Elemental MediaTailor com as plataformas de anúncios do Google

MediaTailor autentica automaticamente as conexões de servidor a servidor com o Google Ad Manager (GAM), o Google Campaign Manager (GCM) e o Google Display & Video 360 (DV360). Quando MediaTailor detecta que uma solicitação de anúncio ou evento de rastreamento é destinado a um servidor de anúncios do Google, ele estabelece automaticamente a conexão segura e autenticada necessária.

Você precisa ter uma conta configurada na plataforma de anúncios relevante do Google. MediaTailor suporta os seguintes tipos de integração:

- **Server-side integração:** protege MediaTailor automaticamente as solicitações de anúncios do lado do servidor e as solicitações de rastreamento de impressões nas plataformas de anúncios do Google. Isso inclui:
 - **Google Ad Manager (GAM):** as solicitações de anúncios para editores ao servidor de anúncios do Google são automaticamente protegidas, o que é necessário para acessar os Compradores Autorizados — o mercado de vendas de anúncios e o intercâmbio de anúncios em tempo real do Google.
 - **Google Campaign Manager (GCM):** Ad-serving e as solicitações de rastreamento são automaticamente encaminhadas pelo endpoint autenticado e seguro do Google, oferecendo suporte aos anunciantes que realizam campanhas nessas plataformas com relatórios mais precisos e menos impressões rejeitadas.
 - **Display & Video 360 (DV360):** as solicitações de anúncios são encaminhadas automaticamente pelo endpoint autenticado e protegido do Google, oferecendo suporte aos anunciantes que realizam campanhas nessas plataformas com relatórios mais precisos e menos impressões rejeitadas.

- Client-side integração: uma integração de jogadores usando o SDK das Bibliotecas de Acesso Programático (PAL) do Google. Essa integração é necessária se você quiser usar o tipo de transação de leilão aberto no Google Ad Manager.

Todas as outras solicitações de anúncios continuam funcionando sem modificações.

O suporte do Ad Manager para tipos de transações programáticas varia de acordo com o tipo de integração que você está usando. Para ver uma lista das opções disponíveis, consulte [Tipos de transação](#) ou entre em contato com a equipe da sua conta do Google.

Tópicos

- [Server-side AWS Elemental MediaTailor integração com as plataformas de anúncios do Google](#)
- [Client-side AWS Elemental MediaTailor integração com o Google Ad Manager](#)

Server-side AWS Elemental MediaTailor integração com as plataformas de anúncios do Google

MediaTailor autentica automaticamente as conexões de servidor a servidor com o Google Ad Manager (GAM), o Google Campaign Manager (GCM) e o Google Display & Video 360 (DV360) usando certificados SSL emitidos pelo Google para. AWS Elemental MediaTailor Quando MediaTailor detecta solicitações de anúncios ou eventos de rastreamento destinados aos servidores de anúncios do Google, ele estabelece automaticamente a conexão segura e autenticada necessária. Nenhuma configuração adicional de certificado SSL ou ticket de suporte é necessária.

Pré-requisito: você deve ter uma conta ativa na plataforma de anúncios relevante do Google (Google Ad Manager, Google Campaign Manager ou Display & Video 360).

MediaTailor aplica a autenticação de certificado Google-issued SSL com base nas seguintes regras de URI de destino:

Google Ad Manager (GAM): solicitações de anúncios do lado do servidor

- Essa regra é acionada em solicitações de anúncios para `serverside.doubleclick.net`, inclusive quando você configura o URL do modelo de ADS com esse domínio ou quando uma solicitação de redirecionamento de anúncio é resolvida para ele.
- MediaTailor adiciona o parâmetro de `ssss=mediatailor` consulta se ele não estiver presente e, em seguida, protege a conexão usando um Google-issued certificado SSL.

- Isso fornece acesso aos compradores autorizados — o mercado de vendas de anúncios em tempo real e a troca de anúncios do Google.

Google Ad Manager (GAM): solicitações de anúncios redirecionadas dentro do VMAP

- Essa regra é acionada em solicitações de anúncios `pubads.g.doubleclick.net` redirecionadas quando você MediaTailor já viu redirecionamentos VAST `server-side.doubleclick.net` na solicitação atual e uma resposta anterior do ADS foi uma resposta do VMAP.
- MediaTailor altera automaticamente o domínio `server-side.doubleclick.net` e estabelece uma conexão autenticada usando o Google-issued certificado SSL.
- Isso permite o acesso a compradores autorizados para fluxos de trabalho de VMAP-based anúncios.

Gerenciador de campanhas do Google (GCM): veiculação de anúncios e solicitações de rastreamento

- Essa regra é acionada em solicitações de `ad.doubleclick.net` veiculação e rastreamento de anúncios quando o URI contém `/ddm/pfadx/` ou `/ddm/trackimp/` (codificado ou não).
- MediaTailor altera automaticamente o domínio `server-side.doubleclick.net` e estabelece uma conexão autenticada usando o Google-issued certificado SSL.
- Isso ajuda os anunciantes a realizarem campanhas nessas plataformas com relatórios mais precisos e menos impressões rejeitadas.

Display & Video 360 (DV360): solicitações de anúncios

- Essa regra aciona solicitações ativadas `bid.g.doubleclick.net` ou de `pagead2.googlesyndication.com` anúncios quando o URI contém `/dbm/vast/` (codificado ou não codificado).
- MediaTailor altera automaticamente o domínio `server-side.doubleclick.net` e estabelece uma conexão autenticada usando o Google-issued certificado SSL.
- Isso permite a veiculação autenticada de anúncios para campanhas do Display & Video 360.

Todas as outras solicitações de anúncios são processadas sem a autenticação do certificado Google-issued SSL.

Note

A autenticação do certificado SSL agora é automática. Se você já enviou um ticket de suporte para habilitar certificados SSL, nenhuma ação é necessária — suas configurações continuam funcionando.

Client-side AWS Elemental MediaTailor integração com o Google Ad Manager

É necessária uma integração MediaTailor do lado do cliente para usar os SDKs das Bibliotecas de Acesso Programático (PAL) do Google Ad Manager e usar o tipo de transação de leilão aberto do Google Ad Manager.

Os SDKs do PAL fornecem informações sobre o conteúdo, o dispositivo e os dados do usuário para uma sessão de reprodução. Por meio do SDK PAL, você pode fornecer essas informações ao Google Ad Manager, que pode então determinar melhor quais anúncios segmentados exibir. Os SDKs estão disponíveis para Android, iOS, HTML5 e Cast. Para obter informações sobre como usar os SDKs PAL, consulte SDK [PAL do Google Ad Manager](#).

Para criar uma integração do lado do cliente com o Ad Manager

1. Use o SDK PAL para gerar um nonce. Para obter informações sobre como configurar um nonce, consulte a documentação do SDK no [Google Ad Manager PAL](#) SDK.

Note

O nonce é uma string criptografada que o PAL gera para solicitações de stream. Cada solicitação deve ter um nome exclusivo.

2. Use o `givn` parâmetro em sua solicitação do ADS para transmitir o valor nonce. Para fazer isso, atualize seu URL do ADS para incluir `&givn=[player_params.givn]`. Para instruções, consulte [Habilitando o rastreamento do lado do cliente](#).

SDKs do reprodutor Datazoom

MediaTailor fez parceria com a Datazoom para fornecer SDKs de jogadores gratuitos para facilitar as integrações com SDKs, como os oferecidos no Google Ad Manager PAL. Para obter informações

sobre o Datazoom e a MediaTailor parceria, consulte. [SDKs de reprodutores gratuitos do Datazoom](#)
[Para acessar os SDKs do player Datazoom, use as informações de contato no site Datazoom with.](#)
[AWS](#)

Personalizando o comportamento de anúncios com a supressão de anúncios

Ao criar uma configuração em AWS Elemental MediaTailor, você pode especificar configurações opcionais de interrupção de anúncios que controlam o comportamento de intervalos de anúncios, incluindo a capacidade de configurar a supressão de anúncios. Isso permite que você personalize as experiências de pausa publicitária do seu conteúdo de vídeo para atender aos seus requisitos específicos.

Restrições de compatibilidade

Você não pode usar a supressão de quebras publicitárias com o seguinte:

- Fluxos de trabalho de VOD e Live-to-VOD. Somente fluxos de trabalho ativos são suportados.
- Server-guided métodos de inserção de anúncios (SGAI). Server-guided os métodos lidam com a tomada de decisões de anúncios de forma diferente e não exigem configuração de supressão.

Tópicos

- [Configurando a supressão de quebras de anúncios](#)

Configurando a supressão de quebras de anúncios

Você pode configurar MediaTailor para ignorar a personalização de anúncios para conteúdo ao vivo. Isso é conhecido como supressão de quebra de anúncio ou supressão de disponibilidade. Este tópico mostra como e também explica como funciona a configuração da supressão de quebras de anúncios.

A supressão de quebras de anúncios pode ser usada para os seguintes casos de uso:

- Janela retroativa de manifesto grande – se um visualizador iniciar a reprodução na borda em tempo real de um manifesto, mas a janela retroativa for grande, convém inserir apenas anúncios iniciados depois que o visualizador começou a assistir. Ou insira anúncios para uma

parte da janela retroativa total no manifesto. Você pode configurar a supressão de anúncios para MediaTailor personalizar intervalos de anúncios em ou dentro de um intervalo de tempo especificado atrás da borda ativa.

- Mid-break participar — se o espectador começar a assistir a uma transmissão de vídeo ao vivo no meio de um intervalo publicitário, é provável que esse usuário mude de canal e não assista ao anúncio. Com a supressão de anúncios, você pode pular a personalização do anúncio se o intervalo do anúncio tiver começado antes de o espectador entrar na transmissão.

Configurando a supressão de anúncios

Para usar a supressão de anúncios, você configura um modo de supressão de disponibilidade, um valor de supressão de disponibilidade e uma política de preenchimento de supressão de disponibilidade das seguintes maneiras:

- No MediaTailor console
- Usando o AWS Command Line Interface (AWS CLI)
- Usando a MediaTailor API ou como parâmetros na solicitação de sessão de reprodução do seu cliente

Para obter informações sobre configuração com parâmetros, consulte [Configurar parâmetros de supressão de anúncios — solicitação de sessão de reprodução](#).

Parâmetros de configuração de supressão de anúncios

Você pode optar por ativar ou desativar a supressão de anúncios. Se você ativar a supressão de anúncios, você especifica se essa supressão ocorre após a borda da reprodução ao vivo ou antes da borda da reprodução ao vivo de uma transmissão ao vivo. Em ambos os casos, você também especifica um horário, em relação à borda ao vivo, em que MediaTailor não personaliza os anúncios. Ao ativar a supressão de disponibilidade, você pode especificar uma política de supressão de disponibilidade que é MediaTailor usada para preenchimentos parciais de intervalos de anúncios quando uma sessão começa no meio do intervalo.

Veja a seguir os parâmetros de configuração de supressão de anúncios:

- Avail suppression mode (Modo de supressão de disponibilidade) – define o modo de supressão de anúncios. Por padrão, a supressão de anúncios fica desativada. Valores aceitos: OFF, BEHIND_LIVE_EDGE, ou AFTER_LIVE_EDGE.

- OFFR: Não há supressão de anúncios e MediaTailor personaliza todos os intervalos publicitários.
- BEHIND_LIVE_EDGE: MediaTailor não personaliza os intervalos publicitários que começam antes da transmissão ao vivo, menos o valor de supressão do Avail. Isso afeta todo o intervalo publicitário, não apenas os anúncios individuais.
- AFTER_LIVE_EDGE: MediaTailor não personaliza os intervalos publicitários que estão dentro da borda ativa, além do valor de supressão do Avail. Isso pode ser configurado para afetar intervalos publicitários inteiros ou permitir o preenchimento parcial dos anúncios.
- Avail suppression value (Valor de supressão de disponibilidade) – um tempo relativo à borda ao vivo em um streaming ao vivo. Accepted value (Valor aceito): um valor de tempo em HH:MM:SS.
- Política de preenchimento de supressão de disponibilidade — Define a política que MediaTailor se aplica ao modo de supressão de disponibilidade. Valores aceitos: PARTIAL_AVAIL, FULL_AVAIL_ONLY.
- BEHIND_LIVE_EDGE modo sempre usa a política de FULL_AVAIL_ONLY supressão.
- AFTER_LIVE_EDGE modo pode ser usado para invocar preenchimentos de intervalos PARTIAL_AVAIL publicitários quando uma sessão começa no meio do intervalo.

Exemplos de configurações de supressão de anúncios

A forma como os [parâmetros de configuração da supressão de anúncios](#) interagem entre si permite especificar várias maneiras diferentes de lidar com a supressão de anúncios e aproveitar o preenchimento antes, durante ou depois da borda ao vivo da transmissão ao vivo. Esta seção fornece exemplos que mostram algumas dessas interações. Use esses exemplos para ajudá-lo a definir os parâmetros de configuração para sua situação específica.

Veja a seguir exemplos de configurações de supressão de anúncios:

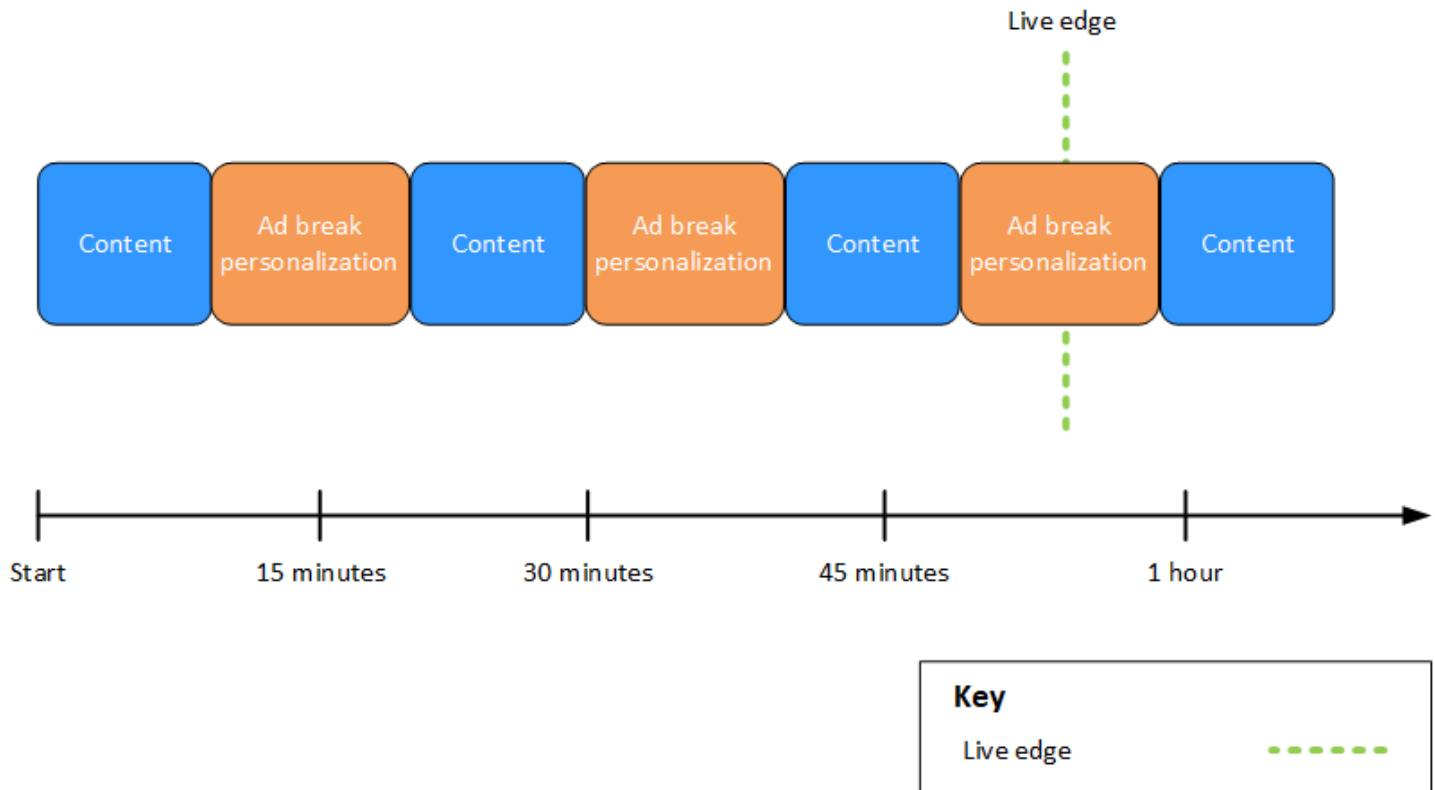
Example 1: Sem supressão de anúncios

Quando o modo de supressão de disponibilidade é OFF, não há supressão de anúncios e MediaTailor personaliza todos os intervalos publicitários.

Na figura a seguir, vários blocos são organizados horizontalmente ao longo de uma linha do tempo que avança da esquerda para a direita. Cada bloco representa uma parte do tempo em que o conteúdo da transmissão ao vivo ou de um intervalo publicitário personalizado é reproduzido. Uma linha pontilhada representa a borda atual da transmissão ao vivo. Dois intervalos publicitários ocorrem antes da borda ao vivo, e outro intervalo de

anúncio está em andamento na borda ao vivo. Conforme mostrado na figura, quando o modo de supressão de disponibilidade está ativado OFF, MediaTailor personaliza todos os intervalos publicitários que ocorrem antes da exibição ao vivo na linha do tempo. MediaTailor também personaliza a pausa publicitária em andamento na borda ao vivo.

Avail suppression mode (default): OFF



Example 2: Supressão de anúncios **BEHIND_LIVE_EDGE** com valor sincronizado com o Live Edge

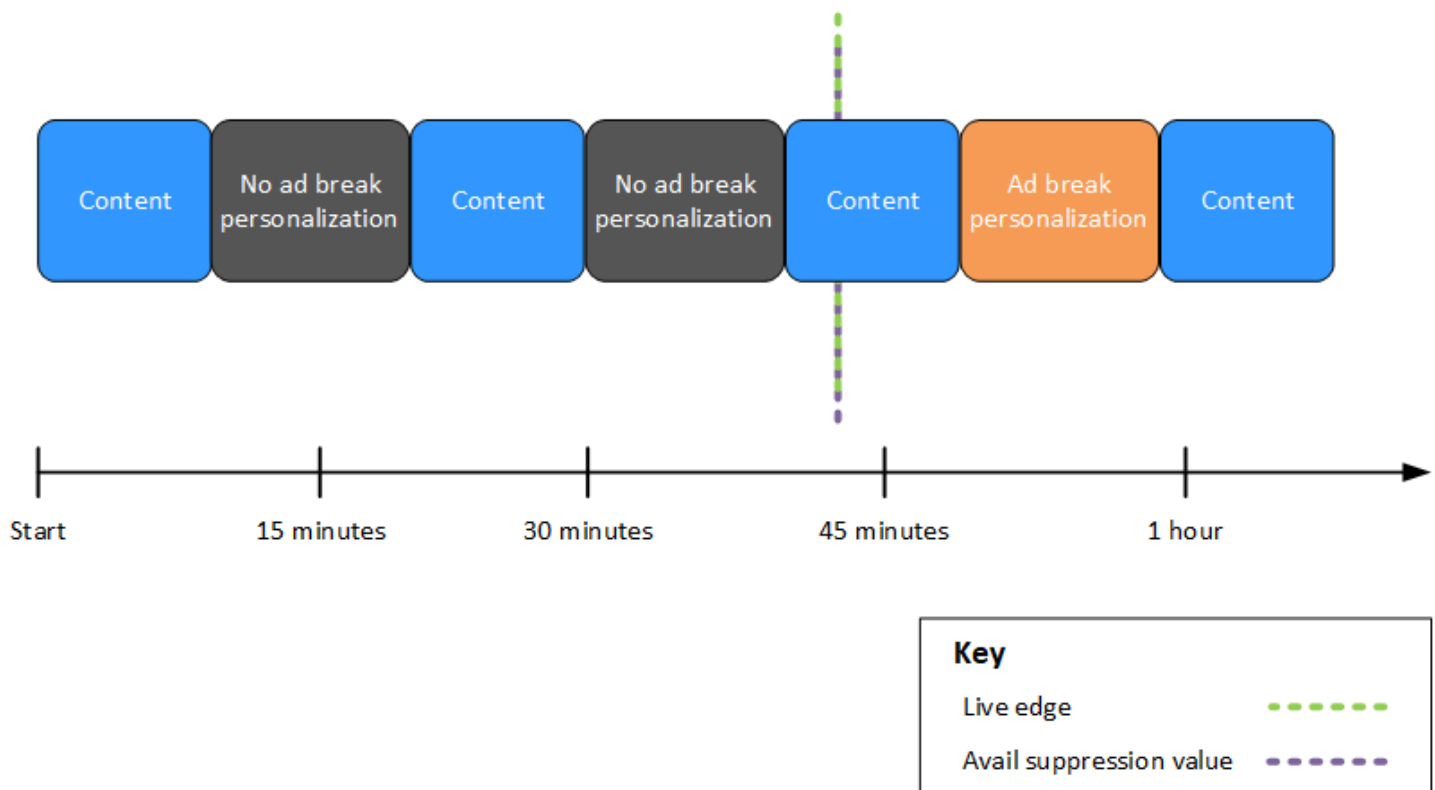
Quando o modo de supressão de disponibilidade está definido como **BEHIND_LIVE_EDGE** e o valor de supressão de disponibilidade está definido como **00:00:00**, o valor de supressão de disponibilidade está sincronizado com a borda ativa. MediaTailor não personaliza nenhum intervalo publicitário que comece antes ou depois da transmissão ao vivo.

Na figura a seguir, vários blocos são organizados horizontalmente ao longo de uma linha do tempo que avança da esquerda para a direita. Cada bloco representa uma parte do tempo em que o conteúdo da transmissão ao vivo, de um intervalo publicitário personalizado ou de um intervalo publicitário não personalizado é reproduzido. Uma linha pontilhada representa a borda atual da transmissão ao vivo. Outra linha pontilhada, representando o valor de supressão de disponibilidade definido como **00:00:00**, se sobrepõe à linha

pontilhada da borda ativa. Dois intervalos de anúncio ocorrem antes da borda ativa e outro ocorre após a borda ativa. Conforme mostrado na figura, quando o modo de supressão de disponibilidade está definido como e o valor de supressão de disponibilidade está definido para que 00:00:00 fique sincronizado com a borda ativa, MediaTailor não personaliza nenhum intervalo publicitário que ocorra antes da borda ativa na linha do tempo. BEHIND_LIVE_EDGE MediaTailor personaliza o intervalo publicitário que ocorre após a transmissão ao vivo.

Avail suppression mode: **BEHIND_LIVE_EDGE**

Avail suppression value: 00:00:00



Example 3: Supressão de anúncios **BEHIND_LIVE_EDGE** com valor por trás do Live Edge

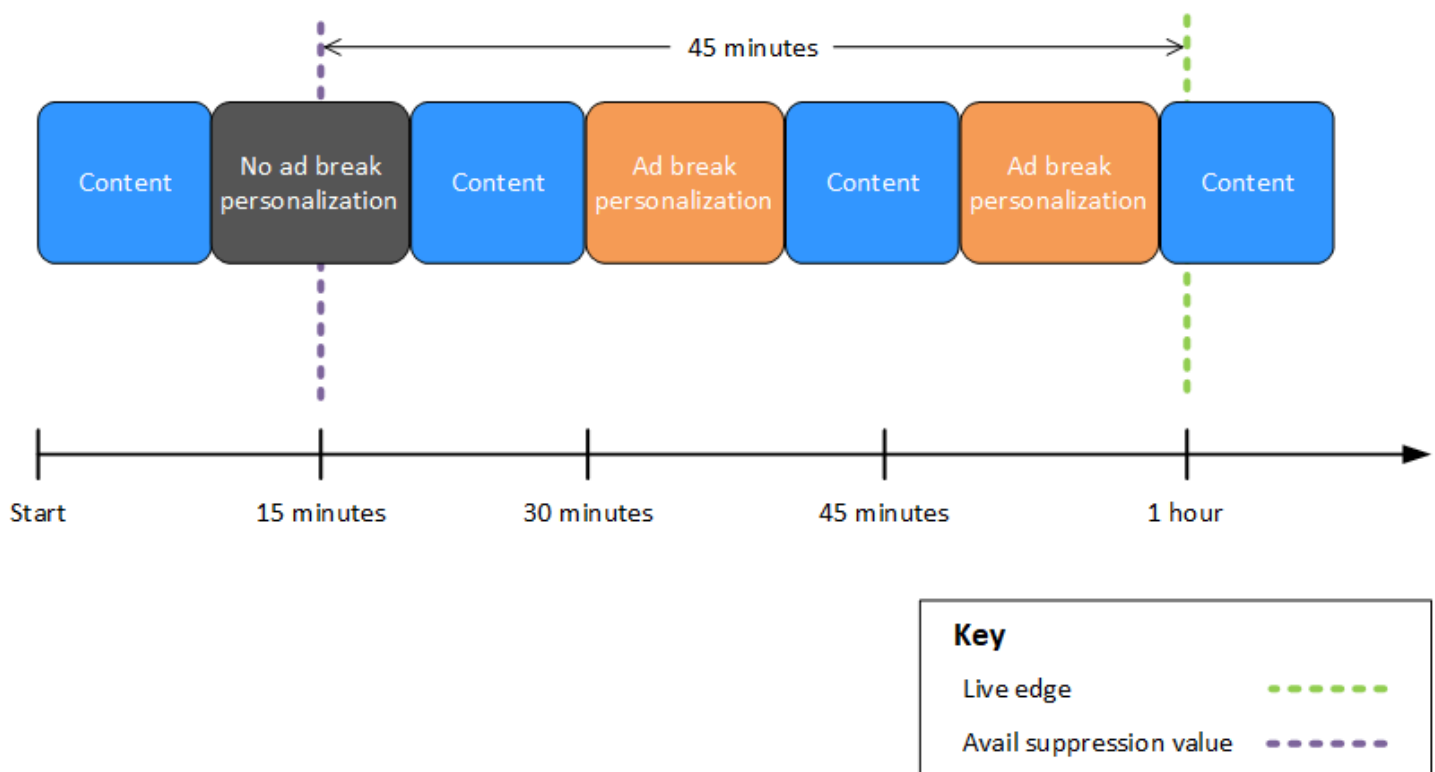
Quando o modo de supressão de disponibilidade está definido como **BEHIND_LIVE_EDGE**, MediaTailor não personaliza nenhum intervalo publicitário antes ou durante esse horário. Neste exemplo, MediaTailor personaliza intervalos publicitários que começam até 45 minutos atrás da borda ao vivo. MediaTailor não personaliza intervalos publicitários que começam mais de 45 minutos atrás da borda ao vivo.

Na figura a seguir, vários blocos são organizados horizontalmente ao longo de uma linha do tempo que avança da esquerda para a direita. Cada bloco representa uma parte do tempo em que o

conteúdo da transmissão ao vivo, de um intervalo publicitário personalizado ou de um intervalo publicitário não personalizado é reproduzido. Uma linha pontilhada representa a borda atual da transmissão ao vivo. Outra linha pontilhada, representando o valor de supressão de disponibilidade definido como `00:45:00`, ocorre 45 minutos antes na linha do tempo em relação à linha pontilhada da borda ativa. O período de 45 minutos entre as linhas pontilhadas representa o período de supressão de disponibilidade. Um anúncio publicitário está em andamento no início do período de supressão de disponibilidade. Dois outros intervalos publicitários ocorrem durante o período de supressão de disponibilidade. Conforme mostrado na figura, quando o modo de supressão de disponibilidade está definido como `BEHIND_LIVE_EDGE` e o valor de supressão de disponibilidade está definido como `00:45:00`, MediaTailor personaliza quaisquer intervalos publicitários que ocorram dentro do período de supressão de disponibilidade. `BEHIND_LIVE_EDGE` MediaTailor não personaliza a pausa publicitária em andamento no início do período de supressão de disponibilidade.

Avail suppression mode: **BEHIND_LIVE_EDGE**

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



Example 4: Supressão de anúncios **AFTER_LIVE_EDGE** sem que ocorram pausas publicitárias durante o período de supressão de disponibilidade

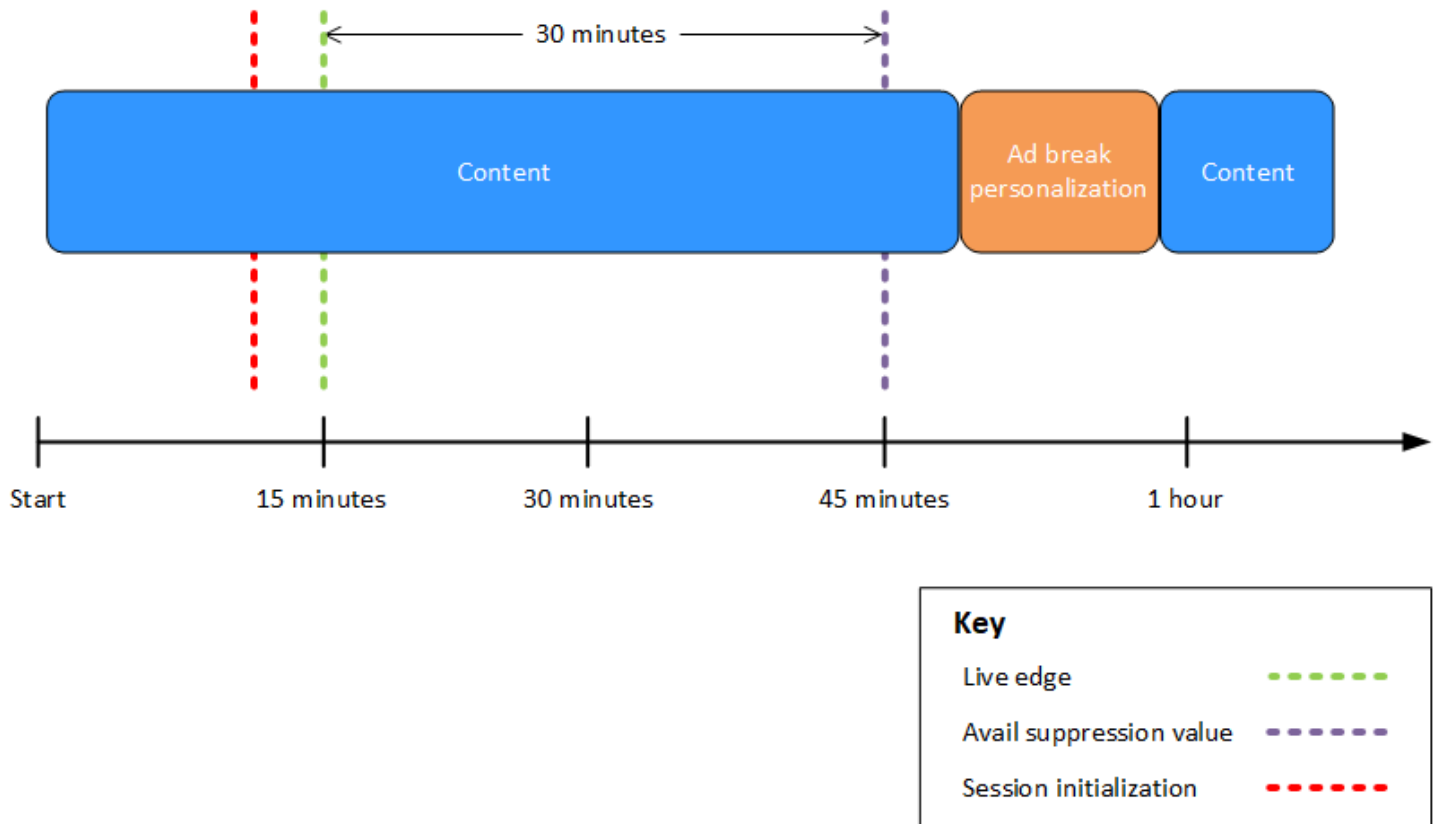
Quando o modo de supressão de disponibilidade está definido como **AFTER_LIVE_EDGE** e o valor de supressão de disponibilidade é maior que zero, MediaTailor não personaliza nenhum intervalo publicitário até que o tempo decorrido da sessão atinja esse valor.

Na figura a seguir, vários blocos são organizados horizontalmente ao longo de uma linha do tempo que avança da esquerda para a direita. Cada bloco representa uma parte do tempo em que o conteúdo da transmissão ao vivo ou de um intervalo publicitário personalizado é reproduzido. Uma linha pontilhada representa a borda atual da transmissão ao vivo. Outra linha pontilhada, representando o valor de supressão de disponibilidade definido como `00:30:00`, ocorre 30 minutos depois na linha do tempo em relação à linha pontilhada da borda ativa. Uma terceira linha pontilhada, representando a inicialização da sessão, ocorre mais cedo na linha do tempo em relação à linha pontilhada da borda ativa. O período de 30 minutos entre o tempo limite ativo e o tempo de supressão de disponibilidade representa o período de supressão de disponibilidade. Um intervalo publicitário ocorre após o período de supressão de disponibilidade.

Conforme mostrado na figura, quando o modo de supressão de disponibilidade está definido como **AFTER_LIVE_EDGE**, o valor de supressão de disponibilidade é definido como `00:30:00` após a borda ativa, e a inicialização da sessão ocorre antes da borda ativa, MediaTailor personalizando quaisquer intervalos publicitários que ocorram após o período de supressão de disponibilidade.

Avail suppression mode: **AFTER_LIVE_EDGE**

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



Example 5: Supressão de anúncios do **AFTER_LIVE_EDGE** com política de preenchimento de **PARTIAL_AVAIL** e uma interrupção de anúncio em andamento no final do período de supressão de disponibilidade

Quando o modo de supressão de disponibilidade está definido como **AFTER_LIVE_EDGE** e o valor de supressão de disponibilidade é maior que zero, MediaTailor não personaliza nenhum intervalo publicitário até que o tempo decorrido da sessão atinja esse valor.

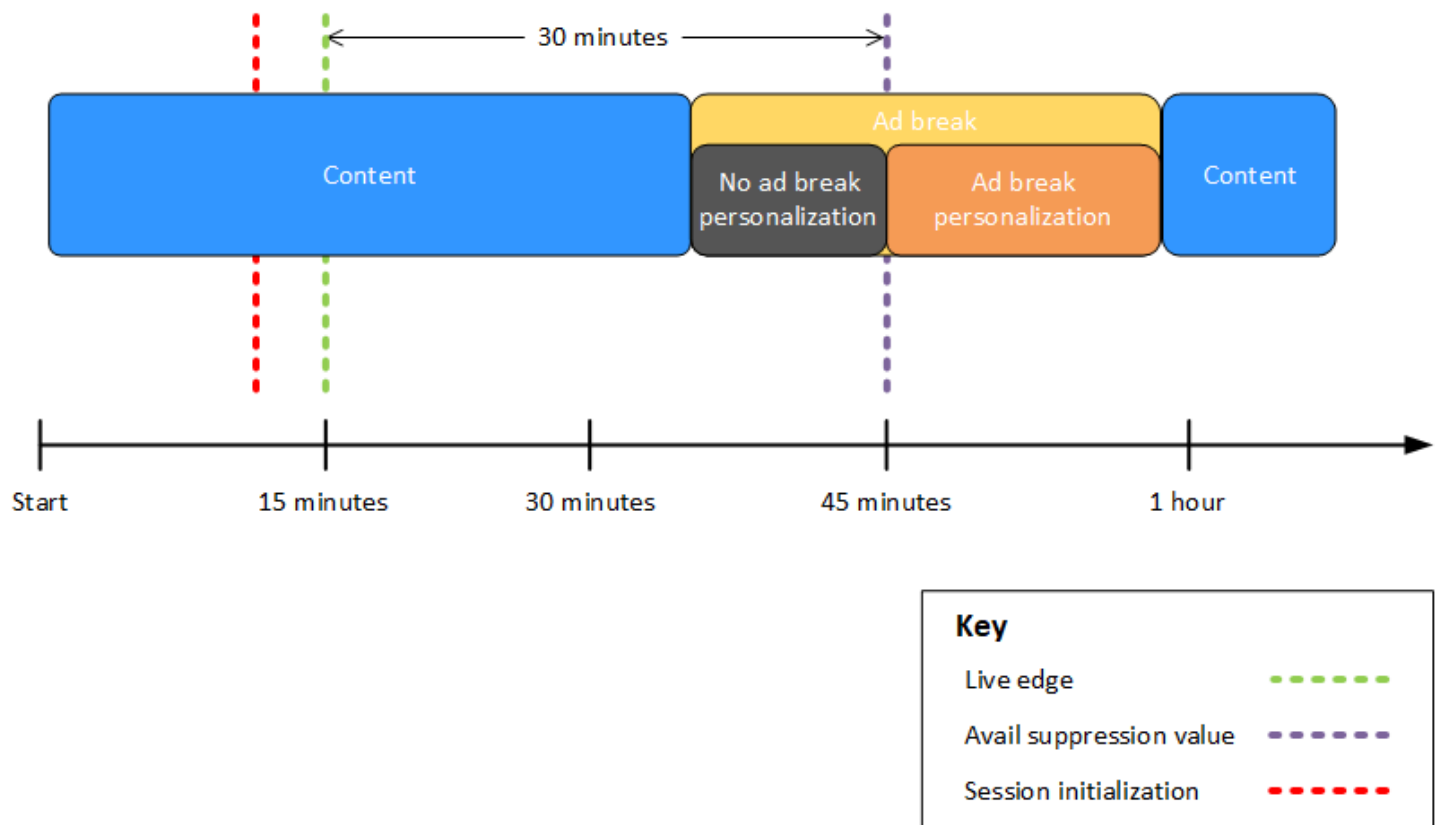
Na figura a seguir, vários blocos são organizados horizontalmente ao longo de uma linha do tempo que avança da esquerda para a direita. Cada bloco representa uma parte do tempo em que o conteúdo da transmissão ao vivo, de um intervalo publicitário personalizado ou de um intervalo publicitário não personalizado é reproduzido. Uma linha pontilhada representa a borda atual da transmissão ao vivo. Outra linha pontilhada, representando o valor de supressão de disponibilidade definido como **00:30:00**, ocorre 30 minutos depois na linha do tempo em relação

à linha pontilhada da borda ativa. Uma terceira linha pontilhada, representando a inicialização da sessão, ocorre mais cedo na linha do tempo em relação à linha pontilhada da borda ativa. O período de 30 minutos entre o tempo limite ativo e o tempo de supressão de disponibilidade representa o período de supressão de disponibilidade. Um anúncio publicitário está em andamento no final do período de supressão de disponibilidade. Conforme mostrado na figura, quando o modo de supressão de disponibilidade está definido como `AFTER_LIVE_EDGE`, o valor de supressão de disponibilidade é definido como `00:30:00` após a borda ativa, a política de preenchimento de supressão de disponibilidade é definida como e a inicialização da sessão ocorre antes da borda ativa `PARTIAL_AVAIL`, MediaTailor personalizando quaisquer intervalos publicitários que ocorram após o período de supressão de disponibilidade. Para o intervalo de anúncio em andamento no final do período de supressão de disponibilidade, MediaTailor personaliza a parte desse intervalo de anúncio que ocorre após o período de supressão de disponibilidade, mas não personaliza a parte desse intervalo de anúncio que ocorre durante o período de supressão de disponibilidade.

Avail suppression mode: `AFTER_LIVE_EDGE`

Avail suppression value: `00:30:00`

Avail suppression fill policy: `PARTIAL_AVAIL`



Example 6: Supressão de anúncios do **AFTER_LIVE_EDGE** com política de preenchimento de **PARTIAL_AVAIL** e uma pausa de anúncio em andamento desde antes da inicialização da sessão até após o final do período de supressão de disponibilidade

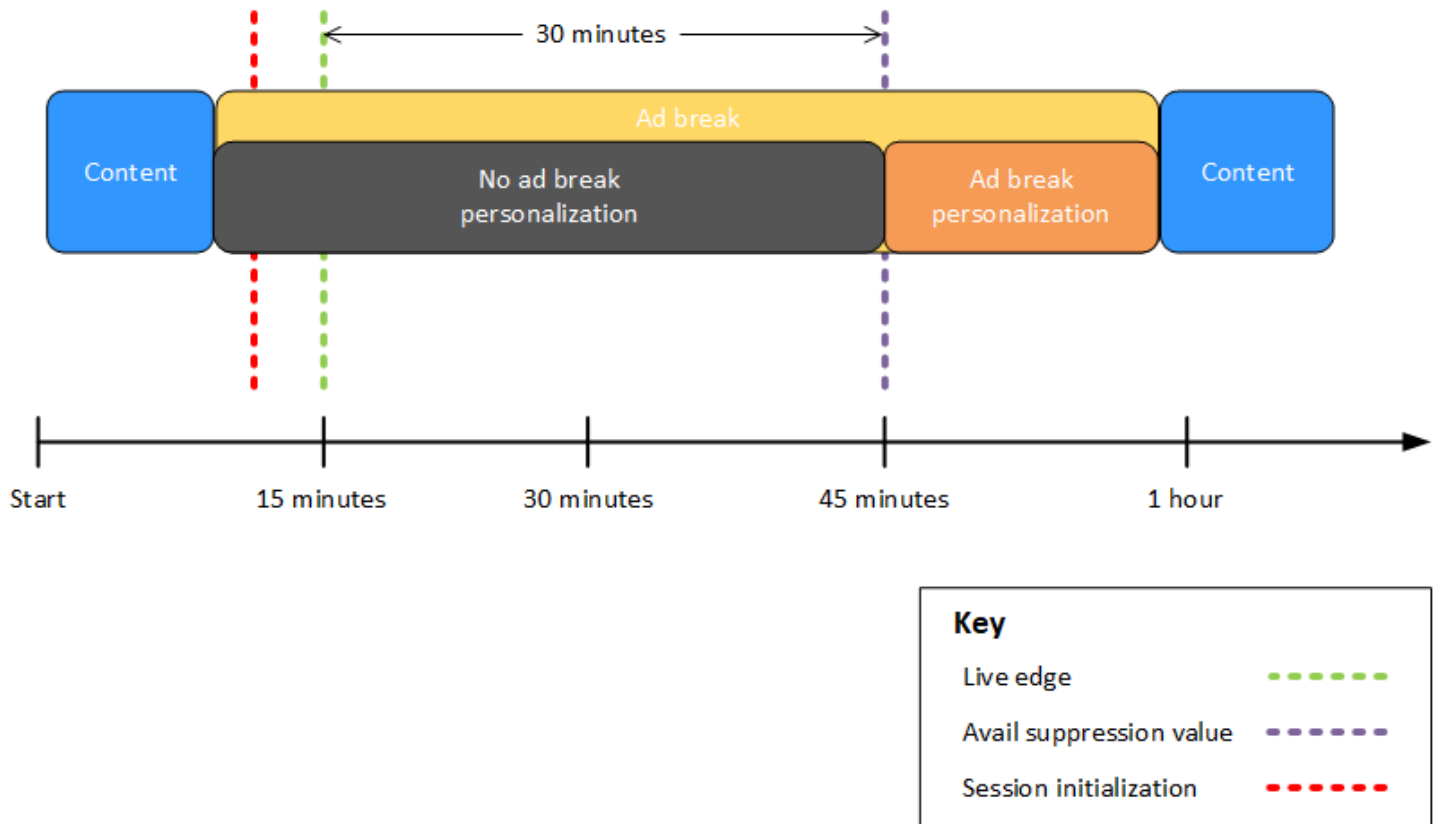
Quando o modo de supressão de disponibilidade está definido como **AFTER_LIVE_EDGE** e o valor de supressão de disponibilidade é maior que zero, MediaTailor não personaliza nenhum intervalo publicitário até que o tempo decorrido da sessão atinja esse valor.

Na figura a seguir, vários blocos são organizados horizontalmente ao longo de uma linha do tempo que avança da esquerda para a direita. Cada bloco representa uma parte do tempo em que o conteúdo da transmissão ao vivo, de um intervalo publicitário personalizado ou de um intervalo publicitário não personalizado é reproduzido. Uma linha pontilhada representa a borda atual da transmissão ao vivo. Outra linha pontilhada, representando o valor de supressão de disponibilidade definido como `00:30:00`, ocorre 30 minutos depois na linha do tempo em relação à linha pontilhada da borda ativa. Uma terceira linha pontilhada, representando a inicialização da sessão, ocorre mais cedo na linha do tempo em relação à linha pontilhada da borda ativa. O período de 30 minutos entre o tempo limite ativo e o tempo de supressão de disponibilidade representa o período de supressão de disponibilidade. Uma pausa publicitária está em andamento desde um momento antes da inicialização da sessão até um momento após o período de supressão de disponibilidade. Conforme mostrado na figura, quando o modo de supressão de disponibilidade está definido como **AFTER_LIVE_EDGE**, o valor de supressão de disponibilidade é definido como `00:30:00` após a borda ativa, a política de preenchimento de supressão de disponibilidade é definida como e a inicialização da sessão ocorre antes da borda ativa **PARTIAL_AVAIL**, MediaTailor personalizando quaisquer intervalos publicitários que ocorram após o período de supressão de disponibilidade. Para o intervalo de anúncio em andamento antes, durante e depois do período de supressão de disponibilidade, MediaTailor personaliza a parte desse intervalo de anúncio que ocorre após o período de supressão de disponibilidade, mas não personaliza a parte desse intervalo de anúncio que ocorre antes ou durante o período de supressão de disponibilidade.

Avail suppression mode: **AFTER_LIVE_EDGE**

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

Avail suppression fill policy: **PARTIAL_AVAIL**



Example 7: Supressão de anúncios **AFTER_LIVE_EDGE** com uma pausa publicitária em andamento no início do período de supressão de disponibilidade

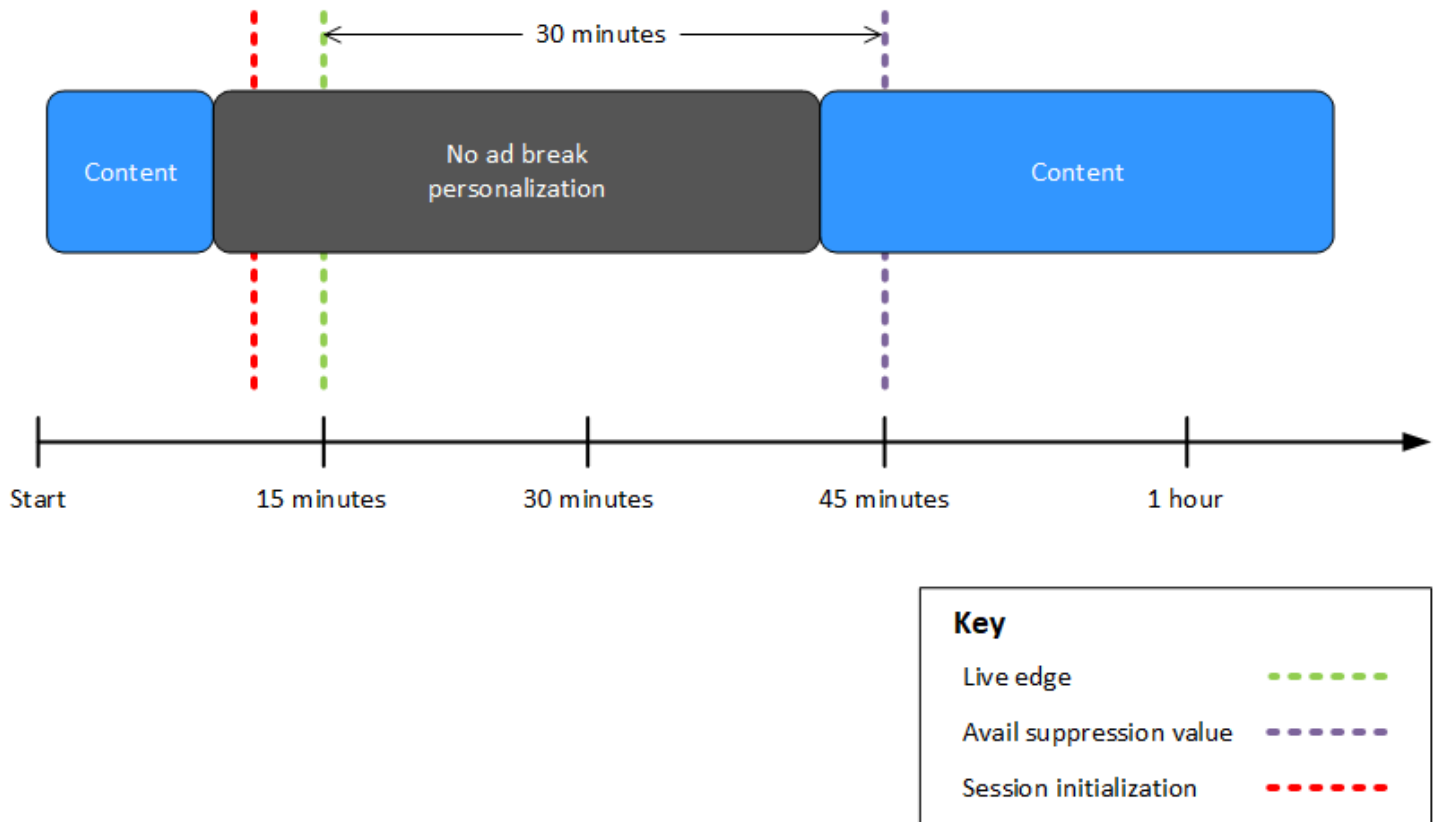
Quando o modo de supressão de disponibilidade está definido como **AFTER_LIVE_EDGE** e o valor de supressão de disponibilidade é maior que zero, MediaTailor não personaliza nenhum intervalo publicitário até que o tempo decorrido da sessão atinja esse valor.

Na figura a seguir, vários blocos são organizados horizontalmente ao longo de uma linha do tempo que avança da esquerda para a direita. Cada bloco representa uma parte do tempo em que o conteúdo da transmissão ao vivo ou de um intervalo publicitário não personalizado é reproduzido. Uma linha pontilhada representa a borda atual da transmissão ao vivo. Outra linha pontilhada, representando o valor de supressão de disponibilidade definido como **00:30:00**, ocorre 30 minutos depois na linha do tempo em relação à linha pontilhada da borda ativa. Uma terceira linha pontilhada, representando a inicialização da sessão, ocorre mais cedo na linha

do tempo em relação à linha pontilhada da borda ativa. O período de 30 minutos entre o tempo limite ativo e o tempo de supressão de disponibilidade representa o período de supressão de disponibilidade. Um intervalo publicitário está em andamento desde um momento antes da inicialização da sessão até um momento dentro do período de supressão de disponibilidade. Conforme mostrado na figura, quando o modo de supressão de disponibilidade está definido como `AFTER_LIVE_EDGE`, o valor de supressão de disponibilidade é definido como `00:30:00` após a borda ativa, e a inicialização da sessão ocorre antes do horário de borda ao vivo, mas após o início do intervalo publicitário, MediaTailor não personaliza esse intervalo publicitário.

Avail suppression mode: `AFTER_LIVE_EDGE`

Avail suppression value: `00:30:00`



Configurar parâmetros de supressão de anúncios — solicitação de sessão de reprodução

Você pode definir as configurações de supressão de anúncios por meio de parâmetros em sua solicitação inicial de sessão de reprodução do lado do servidor ou do lado do cliente para MediaTailor. Se você já definiu as configurações de supressão de anúncios por meio do MediaTailor console ou AWS Elemental MediaTailor da API, esses parâmetros substituem essas configurações.

Tanto o modo de supressão de disponibilidade quanto o valor de supressão de disponibilidade são necessários para que a supressão de anúncios funcione. Esses parâmetros não podem ser configurados de fontes diferentes. Por exemplo, você não pode configurar um parâmetro com o MediaTailor console e outro com um parâmetro de consulta.

MediaTailor suporta os seguintes parâmetros de supressão de anúncios.

Name (Nome)	Description	Valores aceitos
<code>availSuppressionMode</code>	Define o modo de supressão de anúncios. Por padrão, a supressão de anúncios é <code>OFF</code> . Quando definido como <code>BEHIND_LIVE_EDGE</code> , MediaTailor não preenche intervalos publicitários dentro ou fora do <code>aws.availSuppressionValue</code> horário. Quando definido como <code>AFTER_LIVE_EDGE</code> , MediaTailor não preenche intervalos publicitários durante ou após o período de supressão de disponibilidade. O período de supressão do aproveitamento vai do limite de produção até o momento, além de um tempo adicional de <code>aws.availSuppressionValue</code> buffer.	<code>OFF</code> <code>BEHIND_LIVE_EDGE</code> <code>AFTER_LIVE_EDGE</code>
<code>availSuppressionValue</code>	Um tempo relativo a borda ao vivo em um streaming ao vivo.	Um código de UTF-8 URL-encoded tempo inserido <code>HH:MM:SS</code> . Por exemplo, 1 hora e 30 minutos seria <code>01%3A30%3A00</code> .
<code>availSuppressionFillPolicy</code>	Define a política a ser aplicada ao modo de supressão de disponibilidade. <code>BEHIND_LIVE_EDGE</code> sempre usa a política de supressão de disponibilidade total. <code>AFTER_LIVE_EDGE</code> pode ser usado para invocar preenchimentos parciais de intervalos publicitários quando uma sessão começa no meio do intervalo.	<code>PARTIAL_AVAILABLE</code> - não disponível para o modo de <code>BEFORE_LIVE_EDGE</code> supressão

Name (Nome)	Description	Valores aceitos
		FULL_AVAI L_ONLY - o valor padrão para o modo de AFTER_LIV E_EDGE supressão

Server-side configuração

O parâmetro de consulta base é `aws.availSuppression`, que é seguido por pares de nome de parâmetro e valor opcionais. Para criar a consulta, anexe `aws.availSuppression=` ao final da solicitação da sessão de reprodução a MediaTailor, seguido pelos nomes e valores dos parâmetros. Para obter mais informações sobre como criar uma solicitação de sessão de reprodução no lado do servidor, consulte [MediaTailor Rastreamento e relatórios de anúncios do lado do servidor](#).

Exemplo: HLS

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

Server-side a sintaxe da consulta está listada na tabela a seguir.

Componente da string da consulta	Description
?	Um caractere restrito que marca o início de uma consulta.
aws .	A consulta base, que é seguida por parâmetros criados de pares de nome e valor. Para obter uma lista de todos os parâmetros disponíveis, consulte Configurar parâmetros de supressão de anúncios — solicitação de sessão de reprodução .
=	Associa o nome do parâmetro a um valor. Por exemplo, <code>.aws.availSuppressionMode= BEHIND_LIVE_EDGE</code>

Componente da string da consulta	Description
&	Concatena parâmetros de consulta. Por exemplo, <code>aws.availSuppressionMode= BEHIND_LIVE_EDGE</code> e <code>aws.availSuppressionValue= 00:30:00&aws.availSuppressionFillPolicy= FULL_AVAIL_ONLY</code> >.

Client-side configuração

Inclua `availSuppression` parâmetros na solicitação POST do seu cliente para MediaTailor. Para obter mais informações sobre como criar uma solicitação de sessão de reprodução no lado do cliente, consulte [Client-side rastreamento de anúncios](#).

Exemplo: HLS

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

MediaTailor pára-choques e inserção

Os bumpers são cliques de vídeo ou áudio curtos e não puláveis que são reproduzidos no início ou antes do final de um intervalo publicitário. AWS Elemental MediaTailor

As seguintes condições se aplicam aos pára-choques:

- Os amortecedores devem ter 10 segundos ou menos.
- Os amortecedores podem ser inseridos no início de um intervalo publicitário, diretamente antes do final de um intervalo publicitário, ou ambos.
- Os bumpers são reproduzidos durante cada intervalo publicitário em uma sessão de reprodução, a menos que o pre-roll esteja configurado. Se a pré-rolagem estiver configurada, os bumpers

não serão reproduzidos durante o intervalo da pré-rolagem. Em vez disso, eles jogarão em cada intervalo subsequente após o pré-lançamento.

- Para HLS, você deve incluir o `duration` atributo em cada SCTE-35 EXT-X-CUE-OUT tag.
- Os bumpers são transcodificados para corresponder ao conteúdo de origem.
- Você não é cobrado pelos amortecedores.

Configurando amortecedores

Para usar bumpers, configure os URLs dos bumpers com o MediaTailor console, a MediaTailor API ou o AWS Command Line Interface ().AWS CLI Você pode configurar um para-choque inicial, um para-choque final ou ambos. Os bumpers são armazenados em um servidor, como o Amazon Simple Storage Service (Amazon S3). Os URLs do bumper indicam a localização dos ativos do bumper armazenados.

Exemplos de URLs iniciais e finais de bumper:

URL inicial do bumper: `https://s3.amazonaws.com/startbumperad`

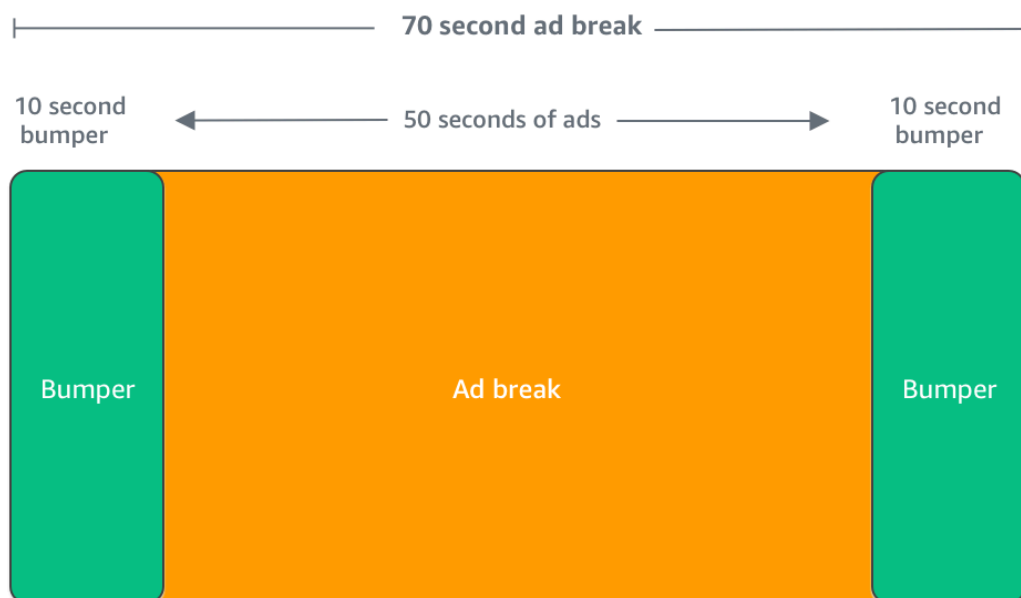
URL final do bumper: `https://s3.amazonaws.com/endbumperad`

Exemplo

Veja a seguir um exemplo do comportamento de anúncios publicitários.

Example Exemplo 1: Para-choques inicial e final

Neste exemplo, os amortecedores inicial e final estão habilitados. O servidor de decisão de anúncios tem 50 segundos de anúncios personalizados para preencher um intervalo publicitário de 70 segundos. O bumper inicial de 10 segundos é reproduzido no início do intervalo publicitário, 50 segundos de exibição do anúncio e, em seguida, o bumper final de 10 segundos.



MediaTailor pré-rol e inserção

Note

Anúncios pré-lançados configuráveis só estão disponíveis para fluxos de trabalho ativos. Para obter detalhes sobre como a inserção de anúncios (incluindo o pré-lançamento) funciona para VOD, consulte [Comportamento de costura de anúncios para VOD](#)

AWS Elemental MediaTailor pode inserir anúncios no início de uma sessão de reprodução, antes do início do conteúdo principal. Esses são chamados anúncios pre-roll.

Para inserir anúncios pre-roll, preencha os campos Live pre-roll ad decision server (Servidor de decisão de anúncio pre-roll) e Live pre-roll maximum allowed duration (Duração máxima permitida do pre-roll em tempo real) nas definições Additional (Adicionais) em sua configuração, conforme descrito em [Configurações opcionais](#).

1. Quando MediaTailor recebe uma solicitação de reprodução, ele envia uma solicitação ao ADS para anúncios pré-lançados com base nos seguintes campos na configuração de MediaTailor reprodução:
 - O servidor de decisão de anúncios pré-lançados ao vivo é o URL do servidor de decisão de anúncios (ADS) que MediaTailor envia a solicitação de anúncios pré-lançados.
 - A duração máxima permitida para pré-lançamento ao vivo é a duração máxima total dos anúncios pré-lançados. MediaTailor executa a seguinte ação com base na duração máxima permitida:
 - Se a duração total dos anúncios na resposta do ADS for menor que o valor que você deu na duração máxima permitida do pré-lançamento ao vivo, MediaTailor insere todos os anúncios. Quando o último anúncio for concluído, ele retornará MediaTailor imediatamente ao conteúdo subjacente.
 - Se a duração total dos anúncios na resposta do ADS for maior do que o valor que você deu na duração máxima permitida para pré-lançamento ao vivo, MediaTailor selecione um conjunto de anúncios que se encaixam na duração sem ultrapassar. MediaTailor insere esses anúncios sem recortes ou truncagens. MediaTailor retorna ao conteúdo subjacente quando o último anúncio selecionado é concluído.
2. Quando MediaTailor recebe a resposta pré-lançada do ADS, ele manipula o manifesto para adicionar links aos anúncios pré-lançados. MediaTailor calcula a hora de início do pré-lançamento e do intervalo da seguinte forma:
 - Para DASH, a fórmula é $(publishTime - availabilityStartTime) - \max(suggestedPresentationDelay, minBufferTime)$.
 - Para HLS, a fórmula é $\max(2 * EXT-X-TARGETDURATION, EXT-X-START:TIMEOFFSET)$.
3. MediaTailor determina a ação a ser tomada em quaisquer intervalos publicitários que não estejam pré-lançados. Se o pré-lançamento se sobrepôr a outro intervalo publicitário, MediaTailor não personaliza a parte sobreposta do intervalo publicitário.

MediaTailor ardósia e inserção

Note

O Slate está disponível apenas para fluxos ao vivo e live-to-VOD de trabalho.

Com AWS Elemental MediaTailor, você pode designar um anúncio tipo slate para intervalos publicitários. Um slate é um MP4 ativo padrão inserido em um stream, como uma imagem estática ou um vídeo em loop, que é reproduzido em vez do conteúdo ao vivo.

AWS Elemental MediaTailor mostra a lista durante os intervalos publicitários nas seguintes situações específicas:

- Para preencher o tempo restante em uma pausa que não foi substituída por anúncios personalizados
- Se o Ad Decision Server (ADS) responder com uma resposta VAST ou VMAP vazia
- Para condições de erro, como tempo limite do ADS ou erro HTTP 500 do ADS
- Se um anúncio não estiver disponível para inserção MediaTailor (por exemplo, quando a transcodificação não foi concluída)

Se você não configurar um slate, usará como MediaTailor padrão o fluxo de conteúdo subjacente quando uma das condições acima for atendida.

Configurando a ardósia

[Você designa a lousa no painel de configuração adicional no console. MediaTailor](#) MediaTailor baixa a lista do URL que você especifica e a transcodifica para as mesmas representações do seu conteúdo. Você pode controlar o tempo máximo em que uma lista será exibida por meio da configuração opcional de limite de personalização no console. MediaTailor Para obter mais informações, consulte [the section called “Como funciona o limite de personalização”](#).

O slate deve ser um MP4 ativo de alta qualidade que contenha áudio e vídeo. Configurar o slate é opcional para configurações não VPAID, mas obrigatório para fluxos de trabalho VPAID.

Note

Caso o servidor que hospeda a barreira use HTTPS, o certificado deve ser de uma autoridade de certificação bem conhecida. Não pode ser um certificado autoassinado. Se você usa um certificado autoassinado, não é AWS Elemental MediaTailor possível recuperar e inserir a lista nos manifestos a partir da origem do conteúdo.

Como funciona o limite de personalização

O limite de personalização define a duração máxima do tempo de anúncio não preenchido (em segundos) permitido em um intervalo publicitário. Esse recurso se aplica especificamente à substituição de anúncios em streams ao vivo e VOD, em vez da inserção de anúncios, pois depende de um fluxo de conteúdo subjacente.

O comportamento varia com base em três cenários:

1. Quando a personalização está desativada:
 - a. O Slate será inserido por toda a duração do tempo não preenchido
 - b. Os amortecedores de início/fim serão inseridos quando configurados (para obter mais informações, consulte) [MediaTailor pára-choques e inserção](#)
 - c. Os anúncios serão inseridos normalmente
2. Quando a personalização está ativada e o limite é menor que a duração do intervalo:
 - a. Se o tempo não preenchido exceder o limite de personalização:
 - i. MediaTailor abandona a personalização do intervalo publicitário
 - ii. O conteúdo subjacente é mostrado
 - iii. Nenhum anúncio, ardósia ou pára-choques são inseridos
 - b. Se o tempo não preenchido for menor que o limite de personalização:
 - i. Anúncios e ardósia são inseridos
 - ii. Os amortecedores são inseridos se configurados
3. Quando a personalização está ativada e o limite é maior que a duração do intervalo:
 - a. Os anúncios serão inseridos
 - b. O Slate será inserido por qualquer tempo restante no intervalo do anúncio
 - c. Os amortecedores serão inseridos se configurados

A tabela a seguir fornece comportamentos detalhados para cenários específicos nos protocolos HLS e DASH:

Matriz de comportamento detalhada

Cenário	Personalização desativada	Personalização ativada em menos do que a duração do intervalo	Personalização habilitada maior do que a duração do intervalo
Esvazie VAST ou VMAP	Ardósia inserida	Nenhuma ardósia inserida	Ardósia inserida
Tempo limite do ADS	Ardósia inserida	Nenhuma ardósia inserida	Ardósia inserida
O anúncio não está disponível (Vast 404)	Ardósia inserida	Nenhuma ardósia inserida	Ardósia inserida
A duração dos anúncios é maior do que o intervalo do anúncio	Anúncios inseridos	Nenhum anúncio inserido	Anúncios inseridos
Tempo de preenchimento que não é totalmente usado por um substituto de anúncio	Ardósia inserida	Se o limite de personalização for maior que o tempo não preenchido: Não ads/slates inserido, caso contrário : Anúncios e lousas inseridos	Ardósia inserida
Pré-rolô com VAST vazio	Ardósia inserida, sem pré-rolô inserido	Nenhuma lousa inserida, nenhum pré-rolô inserido	Ardósia inserida, sem pré-rolô inserido
Pré-lançamento com tempo limite do ADS	Ardósia inserida, sem pré-rolô inserido	Nenhuma lousa inserida, nenhum pré-rolô inserido	Ardósia inserida, sem pré-rolô inserido

Cenário	Personalização desativada	Personalização ativada em menos do que a duração do intervalo	Personalização habilitada maior do que a duração do intervalo
Faça a pré-inscrição quando o anúncio não estiver disponível (Vast 404)	Ardósia inserida, sem pré-rolô inserido	Nenhuma lousa inserida, nenhum pré-rolô inserido	Ardósia inserida, sem pré-rolô inserido
Pára-choques com VAST vazio	Ardósia inserida, pára-choque inserido	Sem ardósias inseridas, sem amortecedores inseridos	Ardósia inserida, pára-choque inserido
Pára-choques com tempo limite de ADS	Ardósia inserida, pára-choque inserido	Sem ardósias inseridas, sem amortecedores inseridos	Ardósia inserida, pára-choque inserido
Amortece quando o anúncio não está disponível (Vast 404)	Ardósia inserida, pára-choque inserido	Sem ardósias inseridas, sem amortecedores inseridos	Ardósia inserida, pára-choque inserido
Amortecedores para preencher um tempo que não é totalmente usado pela substituição de anúncios	Ardósia inserida, amortecedor inserido e inserido	Se o limite de personalização for maior que o tempo não preenchido: Não inserido bumpers/ads/slides inserted, else: No bumpers/ads/slides	Ardósia inserida, amortecedor inserido e inserido

Considerações importantes

Lembre-se do seguinte ao trabalhar com o slate e o limite de personalização.

- Esse comportamento é consistente nos protocolos HLS e DASH.
- O recurso de limite de personalização só se aplica quando o slate é configurado
- Quando o slate estiver configurado e o limite de personalização não estiver configurado, o slate será reproduzido durante todo o período de disponibilidade do Origin ou pelo tempo restante após o preenchimento dos anúncios
- Para anúncios VPAID, MediaTailor insere uma lista com a duração do anúncio VPAID para manter espaço para os anúncios que o player de vídeo inserirá

Configuração do Slate e VPAID

Important

A configuração do Slate é obrigatória ao usar anúncios VPAID. MediaTailor insere o slate para manter espaço para anúncios VPAID que o player de vídeo inserirá. A duração da lista pode ser um pouco maior do que a duração do anúncio VPAID para acomodar a interatividade do usuário.

O player de vídeo então manipula o anúncio VPAID com base nos metadados de relatórios do lado do cliente que MediaTailor retornam, conforme descrito em. [Requisitos do VPAID](#) Para obter informações sobre relatórios no lado do cliente, consulte [the section called “Client-side rastreamento”](#).

Transcodificação de anúncios em AWS Elemental MediaTailor

AWS Elemental MediaTailor transcodifica anúncios, bumpers e slate para corresponder ao formato e às especificações de seus streams de vídeo de origem, garantindo uma reprodução perfeita durante os intervalos publicitários. MediaTailor suporta duas abordagens de transcodificação: transcodificação dinâmica (o padrão) e um perfil de transcodificação personalizado. Compreender as diferenças entre essas abordagens ajudará você a determinar a configuração correta para seus fluxos de trabalho.

Tópicos

- [Transcodificação dinâmica](#)
- [Perfis de transcodificação personalizados](#)

Transcodificação dinâmica

A transcodificação dinâmica é o comportamento padrão de transcodificação de anúncios em MediaTailor. Quando um anúncio, bumper ou slate precisa ser transcodificado, inspeciona MediaTailor automaticamente o manifesto de origem e obtém as configurações de transcodificação em tempo de execução, sem nenhuma configuração de sua parte.

O que a transcodificação dinâmica manipula

A transcodificação dinâmica processa automaticamente os seguintes formulários HLS e DASH fluxos de trabalho.

- Correspondência de codecs padrão — MediaTailor corresponde ao perfil de codec do fluxo de origem. Observe que a correspondência de níveis de codec não é garantida.
- Normalização de áudio — aplica MediaTailor automaticamente a normalização de áudio a anúncios transcodificados usando um nível-alvo padrão de -24 LKFS. Isso garante que os níveis de áudio do anúncio sejam consistentes com seu conteúdo original sem nenhuma configuração adicional. Se seu fluxo de trabalho exigir um destino de normalização de áudio personalizado diferente do padrão de -24 LKFS, é necessário um perfil de transcodificação personalizado.
- Correspondência de comprimento de segmento padrão: por padrão, MediaTailor transcodifica anúncios usando segmentos de 2 segundos. Se seu player precisar de correspondência de comprimento de segmento, você pode ativar a `calculateSegmentLength` bandeira, o que faz MediaTailor com que calcule automaticamente o comprimento correto do segmento analisando os 10 primeiros segmentos da origem e selecionando a duração mais comum como o comprimento do segmento. Essa bandeira está desativada por padrão e deve ser ativada pelo AWS Support. Entre em contato com o AWS Support `calculateSegmentLength` para solicitar a ativação de sua conta ou configurações específicas.
- WebVTT Legendas padrão — Para HLS fluxos de trabalho, MediaTailor preserva as referências do grupo de legendas na playlist multivariante personalizada quando as WebVTT legendas estão presentes no manifesto de origem. Durante os segmentos do anúncio, MediaTailor insere WebVTT segmentos em branco para manter a continuidade da faixa de legendas, porque os criativos de anúncios normalmente não incluem faixas de legendas. Para DASH fluxos de trabalho, MediaTailor exige que os conjuntos de adaptação de legendas no manifesto de origem sejam

segmentados (como vídeo e áudio). Os períodos do anúncio criados por contêm MediaTailor apenas os conjuntos de adaptação fornecidos pelo criativo do anúncio; como os criativos do anúncio normalmente não incluem faixas de legendas, os períodos do anúncio não conterão uma legenda. **AdaptationSet** Os períodos de conteúdo mantêm seus conjuntos de adaptação de legendas normalmente.

- **CMAFsuporte de formato** — MediaTailor detecta se um stream de HLS origem usa CMAF (fMP4) segmentos ou TS segmentos padrão e transcodifica anúncios para que correspondam adequadamente. PoisDASH, o CMAF empacotamento é suportado de forma transparente, pois usa DASH inerentemente ISO-BMFF (fMP4) como formato de contêiner. Nenhuma configuração adicional é necessária para nenhum dos protocolos.

 Note

MediaTailor não suporta HLS fluxos usando HVC1 (HEVC) como codec com TS segmentos. De acordo com a Especificação de Criação HLS para dispositivos Apple, seção 1.5, o formato do contêiner para HEVC vídeo deve ser. fMP4 HLSos fluxos usados HEVC devem, portanto, ser usados CMAF com segmentos MP4 fragmentados.

- **Personalização do Trickplay** — transcodifica MediaTailor automaticamente os anúncios com variantes do Trickplay e fluxos de imagens associados (incluindo I-frame manifestos, miniaturas e fluxos de imagens) para ambos e, de acordo com as especificações do conteúdo de origem. HLS DASH Um perfil de transcodificação personalizado não é mais necessário para esse recurso.
- **DASHManifestos compactos** — suporta MediaTailor automaticamente DASH manifestos compactos, que elevam o SegmentTemplate elemento de Representation elementos individuais para o AdaptationSet nível, reduzindo o tamanho geral do manifesto e melhorando a eficiência da entrega. Um perfil de transcodificação personalizado não é mais necessário para esse recurso.

Quando a transcodificação dinâmica pode não ser suficiente

A transcodificação dinâmica lida com a maioria dos cenários de transcodificação de anúncios. No entanto, há alguns casos em que uma configuração adicional ou um perfil de transcodificação personalizado podem ser necessários.

- **Correspondência de taxa de bits** — A transcodificação dinâmica não garante uma correspondência precisa da taxa de bits por padrão. Se seu fluxo de trabalho exigir que as taxas de bits do anúncio estejam estreitamente alinhadas às taxas de bits de origem, você pode ativar a `enableBitrateRounding` sinalização, que permite um arredondamento da taxa de bits de

até 5% para melhor corresponder aos valores de origem. Essa bandeira deve ser habilitada pelo AWS Support. Entre em contato com o AWS Support `enableBitrateRounding` para solicitar a ativação de sua conta ou configurações específicas. Se seu fluxo de trabalho exigir uma correspondência de taxa de bits acima de 5%, é necessário um perfil de transcodificação personalizado.

- Um alvo de normalização de áudio personalizado diferente do MediaTailor padrão de -24 LKFS
- Non-integer comprimentos de segmento (por exemplo, 1,92s ou 3,84s)
- Áudio Dolby Vision
- Valores específicos de PMT, vídeo ou PID de áudio exigidos por sistemas ou reprodutores downstream
- Taxas de amostragem de áudio personalizadas (por exemplo, 44,1 kHz)
- Proporções de pixels personalizadas (PAR) — por exemplo, 1:1 para pixels quadrados ou 40:33 para tela widescreen NTSC D1/DV
- Faixas de áudio ou faixas de legendas com nomes específicos

Perfis de transcodificação personalizados

Um perfil de transcodificação (CTP) personalizado é uma configuração de trabalho do AWS Elemental de autoria explícita na qual MediaConvert você se registra. MediaTailor Quando um CTP é associado a uma MediaTailor configuração, MediaTailor usa suas configurações definidas para transcodificar cada anúncio, bumper e slate, oferecendo controle preciso sobre codecs, taxas de bits, níveis de normalização de áudio, comprimentos de segmentos, valores de PID, taxas de amostragem e muito mais. Como um CTP substitui completamente a transcodificação dinâmica, você deve configurar explicitamente todos os recursos necessários, incluindo recursos de trickplay (I-frame manifestos, miniaturas e fluxos de imagens), dentro do perfil.

Um CTP é projetado para corresponder a uma configuração de origem específica. Se o conteúdo de origem for diferente da configuração para a qual o CTP foi criado originalmente, os anúncios podem falhar na transcodificação adequada devido a um `NO_VARIANT_MATCH` erro. Isso ocorre quando as saídas CTP não se alinham com as variantes presentes no manifesto de origem. Isso se aplica tanto aos fluxos ao vivo quanto aos VOD fluxos de trabalho. Por exemplo, VOD bibliotecas que contêm conteúdo de fontes diferentes ou codificadas com perfis de codificação diferentes (como resoluções, taxas de bits, codecs ou comprimentos de segmentos variáveis) podem produzir `NO_VARIANT_MATCH` erros se o CTP não considerar todas as combinações de variantes presentes na biblioteca. Se você precisar de um único CTP para oferecer suporte a várias origens ou de uma

VOD biblioteca com perfis de codificação mistos, é possível criar um CTP superconjunto que englobe as variantes de todas as origens de destino. No entanto, os CTPs de superconjuntos estão sujeitos a um limite rígido de 36 variantes de vídeo, o que restringe quantas origens ou combinações de variantes podem ser acomodadas com um único perfil.

 Important

Você deve manter seu CTP sincronizado sempre que fizer alterações nas configurações do codificador ou acessar novas fontes de VOD conteúdo. Não fazer isso resultará em NO_VARIANT_MATCH erros e falhas na transcodificação do anúncio. Isso inclui alterações como adicionar ou remover resoluções, ajustar taxas de bits, alterar codecs, modificar comprimentos de segmentos ou integrar VOD conteúdo de novos provedores com perfis de codificação diferentes.

Ao atualizar seu CTP, esteja ciente do seguinte comportamento.

- A atualização de um CTP existente aplica as novas configurações somente aos anúncios recém-transcodificados. Anúncios que foram transcodificados anteriormente nas configurações antigas de CTP não serão retranscodificados.
- Criar um novo CTP e associá-lo à sua MediaTailor configuração MediaTailor solicitará a retranscodificação de todos os anúncios usando o novo perfil, garantindo a consistência total em todo o seu inventário de anúncios.

Por esse motivo, quando as configurações do codificador mudarem ou você incorporar novas fontes de conteúdo, como prática recomendada, crie um novo CTP em vez de atualizar o existente para garantir que todos os anúncios, incluindo anúncios transcodificados anteriormente, reflitam as configurações atualizadas.

Quando usar um perfil de transcodificação personalizado

Use um perfil de transcodificação personalizado quando seus fluxos de trabalho exigirem um ou mais dos itens a seguir.

- Um alvo de normalização de áudio personalizado diferente do MediaTailor padrão de -24 LKFS
- Non-integer comprimentos de segmento (por exemplo, 1,92 ou 3,84)
- Configurações de áudio Dolby Vision

- Valores específicos de PMT, vídeo ou PID de áudio exigidos por sistemas ou reprodutores downstream
- Taxas de amostragem de áudio personalizadas (por exemplo, 44,1 kHz)
- Correspondência de taxa de bits além do limite de 5% suportado pela transcodificação dinâmica com um sinalizador de configuração
- Proporções de pixels personalizadas (PAR) — por exemplo, 1:1 para pixels quadrados ou 40:33 para tela widescreen NTSC D1/DV
- Faixas de áudio ou legendas com nomes específicos

Introdução aos perfis de transcodificação personalizados

O acesso à API de MediaTailor transcodificação, que é necessário para registrar um CTP, é restrito às contas permitidas. Para começar, entre em contato com o AWS Support usando uma das opções a seguir.

- Solicitar acesso à lista de permissões — Para criar e gerenciar suas próprias CTPs, entre em contato com o AWS Support para solicitar que sua conta seja adicionada à lista de permissões. Depois que sua conta for adicionada à lista de permissões, você terá acesso à API de transcodificação para registrar e gerenciar perfis em qualquer AWS região em que tenha uma configuração existente. MediaTailor
- Solicite a criação de CTP pelo AWS Support — Para que o AWS Support gere e registre um CTP em seu nome, entre em contato com o AWS Support com uma URL de endpoint não DRM-protected original que possa MediaTailor ser acessada. O AWS Support usará esse endpoint para analisar sua configuração de origem e gerar um CTP apropriado para sua conta.

Transcodificação dinâmica versus perfis de transcodificação personalizados

Use a tabela a seguir para determinar a abordagem correta para seu fluxo de trabalho.

Comparação entre transcodificação dinâmica e perfil de transcodificação personalizado

Consideração	Transcodificação dinâmica	Perfil de transcodificação personalizado
Esforço de configuração	Nenhum	Configuração de trabalho do AWS MediaConvert Elemental criada manualmente

Consideração	Transcodificação dinâmica	Perfil de transcodificação personalizado
Suporte para Trickplay (HLS e DASH)	Automático	Deve ser configurado explicitamente no CTP
DASH Manifestos compactos	Automático	Deve ser configurado explicitamente no CTP
CMAF suporte de formato	Automático	Deve ser configurado explicitamente no CTP
Tratamento de alterações de origem	Automático	É necessária uma atualização manual do CTP
Multi-region complexidade	Nenhum	Deve ser registrado por região
Normalização de áudio	-24 LKFS (padrão)	Totalmente configurável
Combinação de taxa de bits	Arredondamento de até 5% via <code>enableBitrateRounding</code> bandeira (entre em contato com o AWS Support para habilitar)	Configurado explicitamente no CTP para corresponder às taxas de bits de origem
Non-integer comprimentos de segmento	Padrão de 2 segundos; comprimentos de segmentos não inteiros não suportados (entre em contato com o AWS Support <code>calculateSegmentLength</code> para habilitar o cálculo automático do comprimento de segmentos inteiros)	Você deve configurar manualmente o comprimento do segmento para corresponder à origem
Valores PID personalizados	Não compatível	Compatível
Áudio Dolby Vision	Não compatível	Compatível

Consideração	Transcodificação dinâmica	Perfil de transcodificação personalizado
Taxas de amostragem de áudio personalizadas	Não compatível	Compatível
Multi-origin apoio	Automatico	É necessário um CTP superdefinido; limitado a 36 variantes de vídeo
MediaTailor melhorias	Automatico	Atualizações manuais de CTP necessárias
Mais adequado para	Correspondência de codecs padrão, perfis de taxa de bits comuns, trickplay, compactos DASH	Requisitos de codificação especializados não atendidos pela transcodificação dinâmica

Pré-busca de anúncios

Use a pré-busca de AWS Elemental MediaTailor anúncios para transmissões ao vivo para ajudar a reduzir o pico de carga nos servidores de decisão de anúncios (ADS) e diminuir a latência de entrega do manifesto no início de cada intervalo de anúncio. Quando você define um cronograma de busca prévia, MediaTailor segue o cronograma para recuperar anúncios do ADS e prepará-los para a inserção de anúncios antes que sejam necessários para um intervalo publicitário. Durante as transmissões ao vivo, a busca prévia pode ajudar a reduzir as taxas de preenchimento de anúncios e a perda de oportunidades de monetização devido a solicitações de anúncios, tempos limite de transcodificação ou outros atrasos na rede.

Note

A busca prévia de anúncios não funciona com métodos de inserção de anúncios guiados pelo servidor (SGAI), incluindo os tradicionais intersticiais guiados pelo servidor e HLS. Os métodos SGAi não exigem busca prévia porque os jogadores buscam apenas os anúncios que vão reproduzir, e os manifestos podem ser veiculados por CDNs sem ver solicitações de sessão individuais. MediaTailor

Para configurar a pré-busca de anúncios, você cria uma ou mais agendas de pré-busca na sua configuração de reprodução. Um cronograma de pré-busca MediaTailor informa como e quando recuperar e preparar anúncios para um próximo intervalo publicitário.

- Se um evento tiver anúncios disponíveis em um cronograma previsível, use um único cronograma de pré-busca. Cada agenda de pré-busca define um único conjunto de anúncios MediaTailor para colocar em um único anúncio disponível. Para pré-buscar anúncios para vários anúncios disponíveis ao usar agendas de pré-busca únicas, você deve criar várias agendas de pré-busca (até 24 horas antes da disponibilidade do anúncio) que se correlacionem com cada disponibilidade do anúncio.
- Se um evento tiver anúncios disponíveis que não estejam em um cronograma previsível, use um cronograma de pré-busca recorrente. Uma agenda de pré-busca recorrente cria automaticamente uma agenda e pré-busca anúncios antes de cada intervalo publicitário em um evento. A programação recorrente de pré-busca recupera anúncios para cada anúncio disponível em um período de tempo definido (até 24 horas antes do término do evento). Você não precisa criar um cronograma para cada anúncio disponível, mas perde parte do controle de tempo que a pré-busca única oferece.

Os tópicos a seguir descrevem mais sobre a busca prévia de anúncios.

Tópicos

- [Como funciona a pré-busca](#)
- [Criação de agendas de pré-busca](#)
- [TPS-based modelagem de tráfego](#)
- [Excluindo agendas de pré-busca](#)

Como funciona a pré-busca

Quando seu cliente faz uma solicitação de manifesto para MediaTailor, o serviço avalia todos os agendamentos de pré-busca associados à configuração de reprodução. Se MediaTailor não encontrar um cronograma de pré-busca correspondente, o serviço reverte para a inserção normal de anúncios e não faz a pré-busca de anúncios.

Se MediaTailor encontrar um cronograma de pré-busca correspondente, o serviço avalia o cronograma com base em dois componentes: recuperação e consumo. A configuração de cada

componente varia entre agendamentos de pré-busca únicos e agendamentos de pré-busca recorrentes, conforme descrito nas seções a seguir.

Fluxo único de agendamento de pré-busca

Recuperação

Isso define a janela de recuperação, que é o intervalo de tempo em que MediaTailor pré-busca anúncios do ADS. Certifique-se de programar essa janela para um horário que seja antes do intervalo publicitário. A seguir, é apresentada uma visão geral de como MediaTailor processa um único cronograma de pré-busca.

Para ver as etapas para criar uma única agenda de pré-busca no console, consulte [Criação de agendas de pré-busca](#). Para obter instruções sobre a API, consulte [PrefetchSchedules](#) na Referência AWS Elemental MediaTailor da API.

Durante a janela de recuperação especificada, MediaTailor envia solicitações ao ADS para recuperar e preparar anúncios para inserção posterior nas sessões de reprodução.

- MediaTailor opcionalmente, usa modelagem de tráfego para limitar o número de solicitações ao ADS de uma só vez. Você pode escolher entre duas abordagens:

Time-window modelagem de tráfego - MediaTailor distribui as solicitações pelo número especificado de segundos em vez de enviar solicitações para todas as sessões ao mesmo tempo. Essa distribuição dispersa de tráfego ajuda a evitar que o ADS fique sobrecarregado, resultando em tempos limite e baixas taxas de preenchimento de anúncios.

TPS-based modelagem de tráfego - MediaTailor limita as solicitações com base em transações por segundo (TPS) e usuários simultâneos. Essa abordagem fornece uma configuração mais intuitiva com base nos limites de capacidade do ADS. Para obter mais informações, consulte [TPS-based modelagem de tráfego](#).

- Se você configurar variáveis dinâmicas, MediaTailor inclua essas variáveis nas solicitações ao ADS. MediaTailor usa essas variáveis para combinar e aproveitar para pré-buscar programações durante a janela de consumo. Consulte a seção Consumo a seguir para obter mais informações.

Example

Um evento ao vivo dura das 7h45 às 10h, com um intervalo publicitário às 8h15. Você configura MediaTailor para recuperar anúncios das 7h45 às 8h, com uma janela de modelagem de tráfego

de 60 segundos. Com 500.000 usuários simultâneos, MediaTailor distribui as solicitações do ADS para atingir uma taxa média de aproximadamente 8.333 transações por segundo por 60 segundos (500.000 users/60 segundos = 8.333 solicitações por segundo), em vez de enviar todas as solicitações simultaneamente.

A configuração de recuperação inclui a chave `scte.event` e o valor 1234 da variável dinâmica. MediaTailor inclui essa variável nas solicitações para o ADS, que então pode ser usada para direcionar anunciantes específicos para a ID de evento 1234.

Consumo

Quando MediaTailor encontra marcadores de quebra de SCTE-35 anúncio durante a janela de consumo, ele coloca os anúncios pré-buscados em um intervalo publicitário.

- Se você não definiu critérios de correspondência de disponibilidade, MediaTailor insere anúncios na primeira pausa na janela de consumo.
- Se você definiu uma chave de variável dinâmica para aproveitar os critérios de correspondência, MediaTailor avalia esses critérios em relação às variáveis dinâmicas definidas na janela de recuperação. Um intervalo publicitário é elegível para inserção de anúncios pré-buscados somente se os critérios de correspondência de disponibilidade forem atendidos. MediaTailor insere anúncios no primeiro intervalo que atendam aos critérios.

Para obter uma lista dos critérios de correspondência disponível compatíveis, consulte a coluna Disponível para pré-busca de anúncios na tabela em: [MediaTailor variáveis de sessão para solicitações de ADS](#)

Exemplo contínuo

Você define a hora de início do consumo como 8h15 e a hora de término como 8h17. Você inclui `scte.event_id` a chave nos critérios de correspondência disponíveis.

Para cada intervalo publicitário que MediaTailor acontece das 8h15 às 8h17, ele avalia o ID do SCTE evento para cada intervalo publicitário. Em cada sessão de reprodução, MediaTailor insere os anúncios pré-buscados no primeiro intervalo publicitário que tem uma ID de evento de 1234 (conforme definido nas variáveis dinâmicas de recuperação). Para intervalos publicitários que não contêm o ID de evento correto, MediaTailor realiza a inserção de anúncios padrão.

Fluxo recorrente do cronograma de pré-busca

Recuperação

Isso define a janela de recuperação recorrente, que é o intervalo de tempo em que MediaTailor pré-busca e insere anúncios para um evento ao vivo (até 24 horas). A seguir, é apresentada uma visão geral de como MediaTailor processa agendamentos de pré-busca recorrentes.

Para ver as etapas para criar uma agenda de pré-busca recorrente no console, consulte [Criação de agendas de pré-busca](#). Para obter instruções sobre a API, consulte [PrefetchSchedules](#) a Referência AWS Elemental MediaTailor da API.

Durante a janela de pré-busca recorrente especificada, MediaTailor recupera e insere anúncios de um evento ao vivo de até 24 horas. Depois de cada intervalo publicitário na janela, recupera MediaTailor automaticamente os anúncios para o próximo intervalo publicitário.

- Se você definir o atraso após o término da disponibilidade, MediaTailor aguarda o tempo especificado antes de recuperar o próximo conjunto de anúncios para o próximo intervalo publicitário.
- MediaTailor opcionalmente, usa modelagem de tráfego para limitar o número de solicitações ao ADS de uma só vez. Você pode escolher entre duas abordagens:

Time-window modelagem de tráfego - MediaTailor distribui as solicitações pelo número especificado de segundos em vez de enviar solicitações para todas as sessões ao mesmo tempo. Essa distribuição dispersa de tráfego ajuda a evitar que o ADS fique sobrecarregado, resultando em tempos limite e baixas taxas de preenchimento de anúncios.

TPS-based modelagem de tráfego - MediaTailor limita as solicitações com base em transações por segundo (TPS) e usuários simultâneos. Essa abordagem fornece uma configuração mais intuitiva com base nos limites de capacidade do ADS. Para obter mais informações, consulte [TPS-based modelagem de tráfego](#).

- Se você configurar variáveis dinâmicas, MediaTailor inclui essas variáveis nas solicitações ao ADS. MediaTailor usa essas variáveis para combinar e aproveitar para pré-buscar programações durante a janela de consumo. Consulte a seção Consumo a seguir para obter mais informações.

Example

Um evento ao vivo dura das 19h às 20h45, com quatro intervalos publicitários durante esse período. Os intervalos publicitários não estão em um cronograma previsível. Você configura

a pré-busca recorrente das 19h às 20h45, com um atraso de 10 minutos e uma janela de modelagem de tráfego de 60 segundos. Depois de cada disponibilidade, MediaTailor recupera os anúncios para o próximo intervalo publicitário. Dez minutos após o término da disponibilidade, MediaTailor começa a enviar solicitações de recuperação para o ADS. Com uma janela de modelagem de tráfego de 60 segundos e 500.000 usuários simultâneos, MediaTailor distribui as solicitações do ADS para atingir uma taxa média de aproximadamente 8.333 transações por segundo durante 60 segundos ($500.000 \text{ users} / 60 \text{ segundos} = 8.333 \text{ solicitações por segundo}$), em vez de enviar todas as solicitações simultaneamente.

A configuração de recuperação inclui a chave `scte.event` e o valor 1234 da variável dinâmica. MediaTailor inclui essa variável nas solicitações para o ADS, que então pode ser usada para direcionar anunciantes específicos para a ID de evento 1234.

Consumo

Quando MediaTailor encontra marcadores de quebra de SCTE-35 anúncio, ele coloca os anúncios pré-buscados em um intervalo publicitário.

- Se você definir a expiração do anúncio recuperado, os anúncios pré-buscados estarão disponíveis para inserção até a expiração especificada.
- Se você não definiu critérios de correspondência de disponibilidade, MediaTailor insere anúncios na primeira pausa na janela de consumo.
- Se você definiu uma chave de variável dinâmica para aproveitar os critérios de correspondência, MediaTailor avalia esses critérios em relação às variáveis dinâmicas definidas na janela de recuperação. Um intervalo publicitário é elegível para inserção de anúncios pré-buscados somente se os critérios de correspondência de disponibilidade forem atendidos. MediaTailor insere anúncios no primeiro intervalo que atenda aos critérios.

Para obter uma lista dos critérios de correspondência disponível compatíveis, consulte a coluna Disponível para pré-busca de anúncios na tabela em [MediaTailor variáveis de sessão para solicitações de ADS](#)

Exemplo contínuo

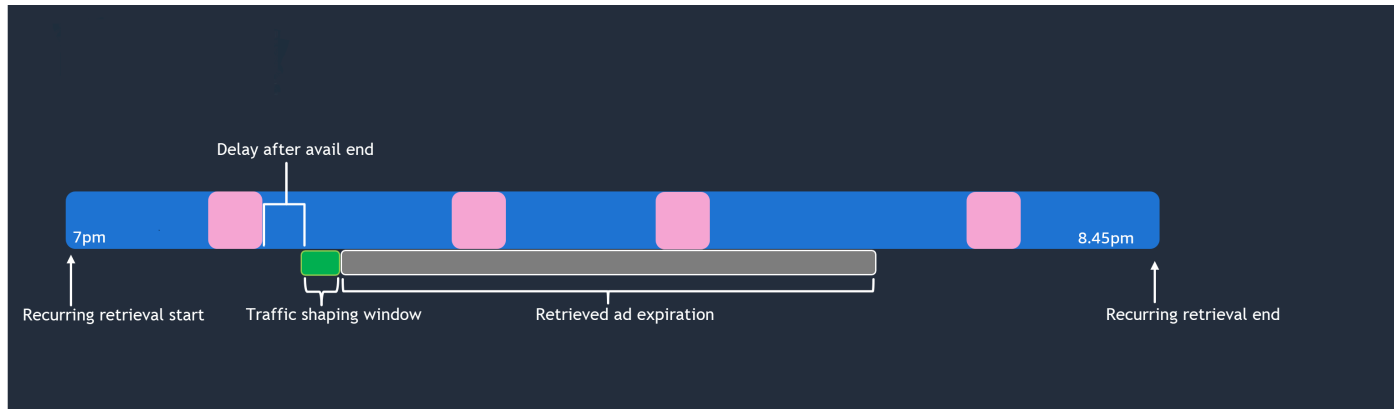
No consumo, você inclui `scte.event_id` a chave nos critérios de correspondência disponíveis.

Para cada intervalo publicitário encontrado MediaTailor, ele avalia o ID do SCTE evento para cada intervalo publicitário. Em cada sessão de reprodução, MediaTailor insere os anúncios pré-buscados em cada intervalo publicitário que tem uma ID de evento de 1234 (conforme definido

nas variáveis dinâmicas de recuperação). Para intervalos publicitários que não contêm o ID de evento correto, MediaTailor realiza a inserção de anúncios padrão.

Você define a expiração do anúncio em 2.700 segundos para que os anúncios recuperados fiquem disponíveis para inserção por 45 minutos.

O gráfico a seguir ilustra o exemplo, com os pequenos quadrados representando intervalos publicitários. As configurações recorrentes do cronograma de pré-busca são ilustradas ao longo da linha do tempo do evento.



Entendendo os custos de pré-busca

Não há custos para fazer solicitações de recuperação de anúncios. No entanto, para a recuperação de anúncios pré-buscados, você será cobrado de acordo com a taxa de transcodificação padrão dos anúncios pré-buscados que transcodificam. MediaTailor Para o consumo de anúncios pré-buscados, você será cobrado pela taxa padrão pela inserção de anúncios para os anúncios pré-buscados colocados em intervalos publicitários. MediaTailor [Para obter informações sobre os custos de transcodificação e inserção de anúncios, consulte Preços.AWS Elemental MediaTailor](#)

Criação de agendas de pré-busca

O procedimento a seguir explica como criar uma agenda de pré-busca usando o MediaTailor console. Para obter informações sobre como criar e gerenciar agendamentos de pré-busca programaticamente usando a MediaTailor API, consulte [PrefetchSchedules](#) a Referência da API.AWS Elemental MediaTailor

 Note

Ao configurar agendamentos de pré-busca MediaTailor, é importante entender como os diferentes tipos de variáveis são tratados.

Aproveite os critérios de correspondência

Se você quiser usar os critérios de correspondência de disponibilidade em uma programação, primeiro configure o modelo de URL do ADS da configuração de reprodução com [variáveis de sessão dinâmicas](#), caso contrário, os critérios de correspondência de disponibilidade não terão efeito. Para obter informações sobre como trabalhar com variáveis dinâmicas de sessão, consulte [Etapa 3: Configurar o URL de solicitação do ADS e os parâmetros de consulta](#) o tópico Introdução à inserção de MediaTailor anúncios.

Variáveis do jogador em agendas de pré-busca

Ao criar um cronograma de pré-busca, não defina as variáveis do player como variáveis dinâmicas na sua configuração de pré-busca. Em vez disso, passe as variáveis do player como você faria normalmente no início da sessão. MediaTailor incluirá automaticamente essas variáveis em solicitações de anúncios pré-buscadas se as variáveis forem mapeadas na URL do modelo do ADS.

Para criar uma nova agenda de pré-busca usando o console

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Configurações. Selecione a configuração de reprodução para a qual você deseja criar uma agenda de pré-busca.
3. Na guia Programações de pré-busca, escolha Adicionar agenda de pré-busca.
4. No painel de detalhes do cronograma do Prefetch, faça o seguinte:
 - Em Nome, insira um identificador para sua agenda de pré-busca, como my-prefetch-schedule.
 - Em Stream ID, opcionalmente, insira um ID exclusivo. Se sua origem contiver vários streams de reprodução, você poderá usar esse ID para MediaTailor instruir a colocar anúncios em um stream específico. Por exemplo, se sua configuração de reprodução tiver um stream de esportes e um stream de um programa de TV, você poderá usar o ID do stream para criar agendas de pré-busca para inserir anúncios segmentados para o stream de esportes. Você

passa o valor do ID do stream para a MediaTailor inicialização da sessão ou solicitação de manifesto do seu cliente. Para obter mais informações, consulte o exemplo a seguir.

- Para rastreamento do lado do servidor, inclua o parâmetro e o valor da `?aws.streamId` consulta na GET HTTP solicitação do seu cliente ao seu endpoint. MediaTailor Para obter informações gerais sobre o rastreamento do lado do servidor, consulte. [MediaTailor Rastreamento e relatórios de anúncios do lado do servidor](#) Uma solicitação de manifesto para um endpoint HLS que inclui um ID de stream tem a seguinte aparência, onde *myStreamId* está o nome do seu ID de stream:

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

- Para o rastreamento do lado do cliente, inclua a `streamId` chave e o valor no corpo da solicitação de inicialização da POST HTTP sessão do seu cliente até o endpoint. MediaTailor/v1/session Para obter informações gerais sobre o rastreamento do lado do cliente, consulte. [Client-side rastreamento de anúncios](#) Uma solicitação de inicialização de sessão que inclui um ID de stream tem a seguinte aparência, onde *myStreamId* está o nome do seu ID de stream:

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

5. Para o tipo de busca prévia, faça sua seleção e escolha a seção correspondente para obter ajuda com campos adicionais:
 - Escolha Único se você estiver criando um cronograma de pré-busca para um intervalo publicitário em um evento.
 - Escolha Recorrente se você estiver criando uma programação que pré-busca anúncios automaticamente antes de cada intervalo publicitário em um evento.

Cronograma único de pré-busca

Para criar uma programação que pré-busque anúncios antes que um anúncio seja disponibilizado em um evento.

1. No painel Recuperação, especifique as configurações de recuperação que você deseja usar. Essas configurações determinam quando MediaTailor pré-busca anúncios do ADS. Eles também determinam quais variáveis dinâmicas de sessão devem ser incluídas na solicitação ao ADS, se houver.
 - Em Hora de início, insira a hora em que MediaTailor você pode iniciar as recuperações prévias para esse intervalo publicitário. MediaTailor tentará pré-buscar anúncios para solicitações manifestas feitas por seu cliente em ou após esse período. O valor padrão é o horário atual. Se você não especificar um valor, o serviço iniciará a recuperação da pré-busca assim que possível.
 - Em Hora de término, insira a hora em que você MediaTailor deseja parar de pré-buscar anúncios para esse intervalo publicitário. MediaTailor tentará pré-buscar anúncios para solicitações de manifesto que ocorrerem nesse momento ou antes dele. A janela de recuperação pode se sobrepor à janela de consumo.
 - Opcionalmente, configure a modelagem de tráfego para limitar o número de solicitações ao ADS de uma só vez. Escolha uma das seguintes abordagens:

Time-window abordagem: em Duração da janela de modelagem de tráfego, insira o número de segundos que MediaTailor devem distribuir as solicitações ao ADS. Para obter mais informações, consulte Explicação da [recuperação do cronograma de pré-busca única](#).

TPS-based abordagem: configure o Peak TPS e o Peak de usuários simultâneos para limitar as solicitações com base nas transações por segundo e nos usuários simultâneos. Para obter mais informações, consulte [TPS-based modelagem de tráfego](#).
 - Na seção [Variáveis dinâmicas](#), insira até 100 variáveis de sessão dinâmica. MediaTailor usa essas variáveis para substituição em solicitações de pré-busca enviadas ao ADS. Se você não inserir nenhuma variável dinâmica de sessão, MediaTailor faça o possível para interpolar os valores das variáveis dinâmicas contidas em seu URL do [ADS](#).
 - Selecione Adicionar variável dinâmica.
 - Em Chave, insira uma chave variável de sessão dinâmica, como `cte.event_id`. Você pode usar qualquer variável dinâmica que MediaTailor ofereça suporte. Para obter informações sobre variáveis dinâmicas de sessão, consulte [MediaTailor variáveis de sessão para solicitações de ADS](#).
 - Em Valor, insira um valor de variável dinâmica, como *my-event*.
 - Para adicionar outra variável dinâmica, escolha Selecionar Adicionar variável dinâmica.

2. No painel Consumo, especifique as configurações que você deseja usar para a janela de consumo. Essas configurações determinam quando MediaTailor os anúncios são colocados no intervalo publicitário. Eles também determinam qualquer critério de correspondência disponível que você queira usar.
 - Em Hora de início, insira a hora em que você MediaTailor deseja começar a colocar anúncios pré-buscados no intervalo publicitário. O valor padrão é a hora atual. Se você não especificar um horário, o serviço iniciará o consumo da pré-busca assim que possível.
 - Em Horário de término, insira um horário em que você MediaTailor deseja parar de colocar os anúncios pré-buscados no intervalo publicitário. MediaTailor tentará pré-buscar anúncios para as solicitações manifestas de seu cliente que ocorram nesse momento ou antes dele. A hora de término deve ser posterior à hora de início e em menos de um dia a partir de agora. A janela de consumo pode se sobrepor à janela de recuperação.
 - Na seção [Disponibilizar critérios](#) de correspondência, selecione Adicionar critérios de disponibilidade e adicione até cinco critérios de correspondência de disponibilidade à sua programação. Em seguida, em Chave de variável dinâmica, adicione uma chave de variável dinâmica, como `scte.event_id`. MediaTailor colocará os anúncios pré-buscados no intervalo publicitário somente se atenderem aos critérios definidos pelos valores das variáveis dinâmicas para MediaTailor as quais seu cliente passa ou que MediaTailor deduzem de informações como dados da sessão. Se um intervalo publicitário não atender aos critérios de correspondência especificados, MediaTailor ignora a busca prévia desse intervalo. Para obter informações, consulte a explicação do [consumo do cronograma de pré-busca única](#).

 Important

Você pode especificar no máximo cinco critérios de correspondência de disponibilidade por programação de pré-busca. Se você especificar vários critérios, MediaTailor combine-os usando um AND lógico. Para modelar combinações lógicas de OR, crie várias agendas de pré-busca.

Use somente variáveis que mudam por intervalo de anúncio (como `scte.event_id`, `scte.segmentation_upid`, ou `avail.index`) como critérios de correspondência disponíveis. Não use variáveis do player (como `player_params.*`) para aproveitar os critérios de correspondência, pois as variáveis do player permanecem constantes durante toda a sessão e não diferenciam os intervalos publicitários. MediaTailor inclui automaticamente as variáveis do player nas

solicitações de anúncios pré-buscadas se as variáveis forem mapeadas na URL do modelo do ADS.

3. Selecione Adicionar critérios de disponibilidade.

Os agendamentos de pré-busca expiram automaticamente após o término da janela de consumo. Para fins de diagnóstico, eles permanecem visíveis por pelo menos 7 dias, após os quais os excluem MediaTailor automaticamente. Como alternativa, você pode excluir manualmente uma agenda de pré-busca a qualquer momento. Para obter informações sobre como excluir manualmente uma agenda de pré-busca, consulte a seção a seguir [the section called “Excluindo agendas de pré-busca”](#).

Determinar com que frequência seu cliente deve chamar a CreatePrefetchSchedule API

Seu cliente pode chamar programaticamente a [CreatePrefetchSchedule](#) API uma vez por dia para configurar a recuperação e o consumo se você souber exatamente quando ocorrerão os intervalos publicitários. Ou seu cliente pode chamar a API várias vezes ao longo do dia para definir a recuperação e o consumo. Ao escolher a frequência de chamadas da API, leve em consideração o [número máximo de agendas de pré-busca ativas](#) e a probabilidade de sua agenda de intervalos publicitários mudar depois de criar suas agendas de pré-busca. Se for provável que o cronograma de intervalos publicitários mude depois que você criar suas agendas de pré-busca, convém chamar a API com mais frequência.

Cronograma de pré-busca recorrente

Para criar uma programação que pré-busque anúncios antes de cada anúncio ser disponibilizado em um evento.

1. No painel Recuperação recorrente, especifique as configurações de recuperação que você deseja usar. Essas configurações determinam quando MediaTailor pré-busca anúncios do ADS. Eles também determinam quais variáveis dinâmicas de sessão devem ser incluídas na solicitação ao ADS, se houver.
 - Em Janela de pré-busca recorrente, insira a hora em que MediaTailor você pode iniciar as recuperações de pré-busca para esse intervalo publicitário. MediaTailor tentará pré-buscar anúncios para solicitações manifestas feitas por seu cliente em ou após esse período. O valor padrão é o horário atual. Se você não especificar um valor, o serviço iniciará a recuperação da pré-busca assim que possível.
 - Em Atraso após o término da disponibilidade, insira o número de segundos que MediaTailor devem esperar após o término de uma disponibilidade antes de pré-buscar anúncios para

a próxima disponibilidade. Se você não especificar um valor, o MediaTailor padrão é sem atraso.

- Opcionalmente, configure a modelagem de tráfego para limitar o número de solicitações ao ADS de uma só vez. Escolha uma das seguintes abordagens:

Time-window abordagem: em Duração da janela de modelagem de tráfego, insira o número de segundos que MediaTailor devem distribuir as solicitações ao ADS. Para obter mais informações, consulte [Explicação da recuperação do cronograma de pré-busca recorrente](#)

TPS-based abordagem: configure o Peak TPS e o Peak de usuários simultâneos para limitar as solicitações com base nas transações por segundo e nos usuários simultâneos. Para obter mais informações, consulte [TPS-based modelagem de tráfego](#).

- Na seção [Variáveis dinâmicas](#), insira até 100 variáveis de sessão dinâmica. MediaTailor usa essas variáveis para substituição em solicitações de pré-busca enviadas ao ADS. Se você não inserir nenhuma variável dinâmica de sessão, MediaTailor faça o possível para interpolar os valores das variáveis dinâmicas contidas em seu URL do [ADS](#).
 - Selecione Adicionar variável dinâmica.
 - Em Chave, insira uma chave variável de sessão dinâmica, como `cte.event_id`. Você pode usar qualquer variável dinâmica que MediaTailor ofereça suporte. Para obter informações sobre variáveis dinâmicas de sessão, consulte [MediaTailor variáveis de sessão para solicitações de ADS](#).
 - Em Valor, insira um valor de variável dinâmica, como *my-event*.
 - Para adicionar outra variável dinâmica, escolha Selecionar Adicionar variável dinâmica.
2. No painel Consumo, especifique as configurações que você deseja usar para a janela de consumo. Essas configurações determinam quando MediaTailor os anúncios são colocados no intervalo publicitário. Eles também determinam qualquer critério de correspondência disponível que você queira usar.
- Em Expiração do anúncio recuperado, indique quanto tempo após a recuperação os anúncios estarão disponíveis para inserção.
 - Na seção [Disponibilizar critérios](#) de correspondência, selecione Adicionar critérios de disponibilidade e adicione até cinco critérios de correspondência de disponibilidade à sua programação. Em seguida, em Chave de variável dinâmica, adicione uma chave de variável dinâmica, como `cte.event_id`. MediaTailor colocará os anúncios pré-buscados no intervalo publicitário somente se atenderem aos critérios definidos pelos valores das variáveis dinâmicas para MediaTailor as quais seu cliente passa ou que MediaTailor deduzem de

informações como dados da sessão. Se um intervalo publicitário não atender aos critérios de correspondência especificados, MediaTailor ignora a busca prévia desse intervalo. Para obter informações, consulte a explicação sobre o [consumo do cronograma de pré-busca recorrente](#).

 Important

Você pode especificar no máximo cinco critérios de correspondência de disponibilidade por programação de pré-busca. Se você especificar vários critérios, MediaTailor combine-os usando um AND lógico. Para modelar combinações lógicas de OR, crie várias agendas de pré-busca.

Use somente variáveis que mudam por intervalo de anúncio (como `scte.event_id`, `scte.segmentation_upid`, ou `avail.index`) como critérios de correspondência disponíveis. Não use variáveis do player (como `player_params.*`) para aproveitar os critérios de correspondência, pois as variáveis do player permanecem constantes durante toda a sessão e não diferenciam os intervalos publicitários. MediaTailor inclui automaticamente as variáveis do player nas solicitações de anúncios pré-buscadas se as variáveis forem mapeadas na URL do modelo do ADS.

3. Selecione Adicionar critérios de disponibilidade.

Os agendamentos de pré-busca expiram automaticamente após o término da janela de consumo. Para fins de diagnóstico, eles permanecem visíveis por pelo menos 7 dias, após os quais os excluem MediaTailor automaticamente. Como alternativa, você pode excluir manualmente uma agenda de pré-busca a qualquer momento. Para obter informações sobre como excluir manualmente uma agenda de pré-busca, consulte a seção a seguir [the section called “Excluindo agendas de pré-busca”](#).

TPS-based modelagem de tráfego

AWS Elemental MediaTailor fornece duas abordagens opcionais de modelagem de tráfego para limitar o número de solicitações ao ADS ao mesmo tempo. TPS-based a modelagem de tráfego oferece uma alternativa à modelagem de tráfego baseada em janela de tempo para agendas de pré-busca. Essa abordagem fornece uma configuração mais intuitiva, permitindo que você especifique a capacidade do servidor de decisão de anúncios (ADS) em termos de transações por segundo (TPS) e de usuários simultâneos esperados, em vez de cálculos de tempo.

Como funciona a modelagem de TPS-based tráfego

Em vez de especificar as durações da janela de recuperação, você fornece os seguintes parâmetros:

Pico de TPS

O número máximo de solicitações por segundo que seu ADS pode processar. Esse parâmetro não tem valor padrão.

Pico de usuários simultâneos

O número máximo esperado de espectadores simultâneos para seu conteúdo. Esse parâmetro não tem valor padrão.

MediaTailor distribui automaticamente as solicitações de pré-busca ao longo do tempo para permanecer dentro do limite de TPS especificado, independentemente do número de sessões simultâneas.

Exemplo TPS-based exemplo de configuração

Seu ADS pode lidar com 500 TPS e você espera 100.000 espectadores simultâneos durante os horários de pico. Você configura:

- Pico de TPS: 500
- Pico de usuários simultâneos: 100.000

MediaTailor distribui automaticamente as solicitações de pré-busca ao longo do tempo para permanecer dentro do limite de TPS especificado, independentemente do número de sessões simultâneas.

Excluindo agendas de pré-busca

O procedimento a seguir explica como excluir uma agenda de pré-busca usando o MediaTailor console. Para obter informações sobre como excluir agendamentos de pré-busca programaticamente usando a MediaTailor API, consulte [DeletePrefetchSchedule](#) a Referência da API.AWS Elemental MediaTailor

 Note

A exclusão não ocorre em tempo real. Você pode enfrentar um atraso ao MediaTailor excluir os agendamentos de pré-busca, período durante o qual a recuperação e o consumo da pré-busca continuarão sendo executados em segundo plano.

Para excluir uma agenda de pré-busca usando o console

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Configurações. Selecione a configuração de reprodução que contém os agendamentos de pré-busca que você deseja excluir.
3. Na guia Agendas de pré-busca, selecione a agenda de pré-busca que você deseja excluir. Em seguida, selecione Excluir.

Usando anúncios pré-condicionados com AWS Elemental MediaTailor

Em um [fluxo de trabalho típico de inserção de anúncios](#), transcodifica MediaTailor dinamicamente os anúncios para que correspondam ao fluxo de conteúdo, os salva e une os anúncios à transmissão ao vivo. Como esse processo ocorre somente depois de MediaTailor receber o anúncio em uma resposta VAST do servidor de decisão de anúncio (ADS), há um atraso no momento em que o anúncio está disponível para edição. Se uma latência adicional for introduzida no fluxo de trabalho de criação de anúncios (seja devido ao tempo limite do ADS ou a outros problemas de conteúdo ou rede), MediaTailor poderá preencher parcialmente a disponibilidade ou perder completamente o intervalo do anúncio.

Para reduzir o tempo necessário para incluir anúncios em seu conteúdo, você pode usar anúncios pré-condicionados. Um anúncio pré-condicionado é aquele que você transcodifica antes de usá-lo na inserção de MediaTailor anúncios. Em vez de fornecer URLs os anúncios não condicionados para o seu ADS, você fornece URLs os anúncios pré-condicionados. Em sua resposta VAST à MediaTailor solicitação, o ADS inclui links diretos para os anúncios pré-condicionados. Ao remover a parte de transcodificação da costura do anúncio, MediaTailor basta salvar o anúncio e juntá-lo ao fluxo de conteúdo. O processo de criação de anúncios com anúncios pré-condicionados reduz o tempo entre o momento em que o anúncio MediaTailor é informado por meio da resposta VAST e o momento em que o anúncio é incorporado ao conteúdo.

Como alternativa, você também pode usar a pré-busca de anúncios, que é quando você configura MediaTailor para realizar o processo de criação de anúncios em um horário agendado antes que o intervalo seja necessário. Para obter mais informações sobre a pré-busca de anúncios, consulte.

[Pré-busca de anúncios](#)

Requisitos de anúncios pré-condicionados

A seguir estão os requisitos a serem considerados ao configurar um fluxo de trabalho de criação de anúncios com anúncios pré-condicionados.

Requisitos do **MediaFiles**

A resposta VAST para a qual o servidor de anúncios envia MediaTailor deve incluir MediaFiles que atendam aos seguintes requisitos:

O anúncio (Creative) deve ter variantes que estejam em conformidade com as variantes de taxa de bits do fluxo de conteúdo. É sua responsabilidade garantir que a resposta do VAST use as variantes de anúncio certas para corresponder aos manifestos do modelo.

Embora o uso de anúncios pré-condicionados possa ajudar a tornar a inserção de anúncios mais eficiente, MediaTailor não tem a capacidade de gerenciar o processo de transcodificação para garantir que os arquivos de mídia dos anúncios sejam compatíveis com as especificações do manifesto de conteúdo. Se o anúncio não corresponder ao fluxo de conteúdo, a inserção MediaTailor pode ser perdida ou a incompatibilidade pode causar erros no dispositivo de reprodução.

Além disso, para ser incorporado ao fluxo de conteúdo sem MediaTailor transcodificação, é MediaFile necessário atender aos seguintes requisitos:

- Ele deve estar acessível na Internet pública para que MediaTailor você possa baixá-lo.
- Ele deve usar a entrega de streaming, indicada como `delivery="streaming"` na resposta VAST.
- Ele deve ser um arquivo `.m3u8` (para HLS) ou `.mpd` (para DASH).

Example Resposta VAST

No exemplo de resposta VAST a seguir, MediaTailor insira o MediaFile com o seguinte: URLs

- Para um fluxo HLS, MediaTailor usa `https://example-ad-origin.amazonaws.com/ad1/index_low.m3u8`. Este é o primeiro MediaFile com entrega de streaming e uma extensão de arquivo compatível (`.m3u8`).

- Para um stream do DASH, MediaTailor usa `https://example-ad-origin.amazonaws.com/ad1/index.mpd`. Este é o primeiro MediaFile com entrega de streaming e uma extensão de arquivo compatível (. mpd).

```
<?xml version="1.0" encoding="UTF-8"?>
<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="3.0">
  <Ad id="ad1">
    <InLine>
      <AdSystem>ExampleAdSystem</AdSystem>
      <AdTitle>ad1</AdTitle>
      <Impression><![CDATA[https://example-impression.amazonaws.com]]></
Impression>
      <AdServingId>de8e0d33-9c72-4d77-bb3a-f7e566ffc605</AdServingId>
      <Creatives>
        <Creative id="creativeId1" sequence="1">
          <Linear skipoffset="00:00:05">
            <Duration>00:00:30</Duration>
            <MediaFiles>
              <MediaFile delivery="progressive" width="1280" height="720"
type="video/mp4" bitrate="533" scalable="true" maintainAspectRatio="true"><![
CDATA[https://example-ad-origin.amazonaws.com/ad1/ad1.mp4]]></MediaFile>
              <MediaFile delivery="streaming" width="1280"
height="720" type="application/dash+xml" bitrate="533" scalable="true"
maintainAspectRatio="true"><![CDATA[https://example-ad-origin.amazonaws.com/ad1/
index.mpd]]></MediaFile>
              <MediaFile delivery="streaming" width="640"
height="360" type="application/x-mpegURL" bitrate="262" scalable="true"
maintainAspectRatio="true"><![CDATA[https://example-ad-origin.amazonaws.com/ad1/
index_low.m3u8]]></MediaFile>
              <MediaFile delivery="streaming" width="2560"
height="1440" type="application/x-mpegURL" bitrate="1066" scalable="true"
maintainAspectRatio="true"><![CDATA[https://example-ad-origin.amazonaws.com/ad1/
index_high.m3u8]]></MediaFile>
            </MediaFiles>
          </Linear>
        </Creative>
      </Creatives>
    </InLine>
  </Ad>
</VAST>
```

Requisitos do manifesto de anúncios

Para usar anúncios pré-condicionados, seus manifestos de anúncios para pais e filhos devem atender aos seguintes requisitos:

- O manifesto vinculado na *Creative* seção da resposta VAST deve ser o manifesto do anúncio principal.
- URLs Para a criança, os manifestos de anúncios devem ser caminhos relativos.
- Os manifestos publicitários secundários devem estar no mesmo diretório da playlist multivariante principal, no mesmo nível. Os manifestos infantis não podem estar em um subdiretório ou em outro local.

Example playlist multivariante parental suportada

A playlist multivariante de anúncios principais a seguir contém playlists de mídia publicitária relativas URLs para crianças. As playlists secundárias também estão no mesmo diretório da playlist multivariante principal.

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

Example playlist multivariante principal não suportada: subdiretórios

A seguinte playlist multivariante de anúncios principais contém playlists secundárias que estão em subdiretórios em relação à playlist multivariante principal. Não é uma playlist compatível com anúncios pré-condicionados.

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

Example playlist multivariante principal não suportada: absoluta URLs

A seguinte playlist multivariante de anúncios principais contém playlists secundárias com valores absolutos. URLs Não é uma playlist compatível com anúncios pré-condicionados.

```
#EXTM3U
#EXT-X-STREAM-INF:BANDWIDTH=150000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
https://example.mediatailor.us-west-2.amazonaws.com/index_1.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=440000,RESOLUTION=416x234,CODECS="avc1.42e00a,mp4a.40.2"
https://example.mediatailor.us-west-2.amazonaws.com/index_2.m3u8
#EXT-X-STREAM-INF:BANDWIDTH=640000,RESOLUTION=640x360,CODECS="avc1.42e00a,mp4a.40.2"
https://example.mediatailor.us-west-2.amazonaws.com/index_3.m3u8
```

Fluxo de trabalho de anúncios pré-condicionados

A seguir está uma descrição básica de como os anúncios pré-condicionados funcionam em um fluxo de trabalho de costura de anúncios com. MediaTailor A primeira parte do fluxo de trabalho são as ações que você deve realizar para se preparar para usar anúncios pré-condicionados. A segunda parte descreve como MediaTailor processa os anúncios.

Parte 1: configuração de anúncios pré-condicionados

Conclua as etapas a seguir para configurar um fluxo de trabalho que usa anúncios pré-condicionados em MediaTailor.

1. Use um serviço de transcodificação, como AWS Elemental MediaConvert, para condicionar seus criativos em variantes que suportem as diferentes taxas de bits, resoluções e codecs de seus manifestos de modelo.
2. Forneça URLs os arquivos de mídia pré-transcodificados para seu ADS, para uso em respostas VAST.
3. [Crie sua configuração de reprodução](#) em MediaTailor. Para usar anúncios pré-condicionados, selecione Nenhum para a configuração de condicionamento do arquivo de mídia de streaming na configuração.
4. Continue com a entrega de conteúdo configurada normalmente.

Parte 2: processamento de MediaTailor anúncios

MediaTailor e a costura é concluída conforme descrito em. [Como funciona a inserção de MediaTailor anúncios](#) Quando MediaTailor recebe uma resposta VAST do ADS, ele usa a lógica a seguir

para determinar quais ações tomar para os anúncios. Essa lógica é ditada pela configuração de condicionamento do arquivo de mídia de streaming na configuração de reprodução.

- Quando o condicionamento do arquivo de mídia de streaming está definido como Transcodificar, MediaTailor transcodifica os arquivos de mídia com a progressive entrega e os une ao manifesto. Se não houver anúncios suficientes com arquivos de mídia progressive de entrega para preencher a disponibilidade, MediaTailor transcodifica e usa esses anúncios com streaming entrega.
- Quando o condicionamento de arquivos de mídia de streaming está definido como Nenhum MediaTailor, agrupa anúncios com arquivos streaming de mídia de entrega no manifesto sem transcodificá-los. Se não houver anúncios suficientes com arquivos de mídia streaming de entrega para preencher a disponibilidade, MediaTailor transcodifica e usa esses anúncios com progressive entrega.

Trabalhando com funções de MediaTailor monetização

Com as Funções de MediaTailor Monetização (Funções), você pode personalizar a forma como AWS Elemental MediaTailor gerencia os dados da sessão e cria solicitações de anúncios durante a inserção do anúncio. As funções permitem chamar APIs externas, transformar dados com expressões e modificar parâmetros de solicitação do servidor de decisão de anúncios (ADS) ou parâmetros do player. Você não precisa implantar ou gerenciar uma infraestrutura personalizada.

As funções usam JSonata, uma linguagem leve de consulta e transformação para dados JSON, para avaliar expressões. Você escreve expressões JSonata para ler dados da sessão, transformar valores e definir a saída. Para obter a lista completa das funções JSonata suportadas, consulte [Referência de expressão JSonata](#)

Use Funções quando precisar:

- Enriqueça as solicitações de anúncios com dados do espectador. Ligue para um serviço de identidade no início da sessão e inclua a identidade resolvida em cada solicitação do ADS para segmentação personalizada de anúncios.
- Personalize os parâmetros de solicitação do ADS. Defina dinamicamente o URL, os cabeçalhos ou o corpo do ADS com base nos dados da sessão, nos SCTE-35 sinais ou nas respostas externas da API.
- Execute A/B testes em servidores de anúncios. Divida o tráfego entre diferentes endpoints do ADS com base nos atributos da sessão ou na atribuição aleatória.

- Crie tubulações de enriquecimento em várias etapas. Encadeie várias funções para buscar dados de uma API, transformá-los e passar os resultados para a próxima etapa.

Como as funções se encaixam no MediaTailor fluxo de trabalho

Quando um espectador inicia uma sessão de reprodução, MediaTailor avalia suas funções em pontos específicos do fluxo de inserção de anúncios, chamados de ganchos do ciclo de vida.

1. Você define uma função e a anexa a uma configuração de reprodução por meio de um mapeamento de funções.
2. MediaTailor executa a função no gancho de ciclo de vida designado.
3. A função lê os dados da sessão, opcionalmente chama uma API externa e grava os resultados que são MediaTailor usados durante o processamento da reprodução.

Tip

Se você é iniciante em Funções, comece com o [Guia de início rápido](#) e depois leia [Hooks do ciclo de vida](#). Para obter a sintaxe da expressão e exemplos completos, consulte [Referência de expressão JSonata](#) e [Exemplos de funções](#)

Tópicos

- [Guia de início rápido do Functions](#)
- [Ganchos de ciclo de vida de funções](#)
- [Tipos de funções e composição](#)
- [Criação e gerenciamento de funções](#)
- [Referência de expressão JSonata para Funções](#)
- [Exemplos de funções](#)
- [Funções de solução de problemas e monitoramento](#)
- [Limites de funções](#)

Guia de início rápido do Functions

Este guia explica como criar sua primeira função, anexá-la a uma configuração de reprodução e verificar se ela foi executada. Ao final, você terá uma função funcional que classifica o tipo de dispositivo de cada espectador (ctv,mobile, oudesktop) e o armazena como um parâmetro de player disponível em cada solicitação do ADS.

Pré-requisitos

Antes de começar, verifique se você tem uma configuração de MediaTailor reprodução existente. Se você não tiver uma, consulte [Começando com MediaTailor](#).

Etapa 1: criar uma função do

Nesta etapa, você cria uma função que classifica o tipo de dispositivo do visualizador com base na string do agente do usuário e armazena o resultado nos parâmetros do player. A função usa um tipo de saída personalizado (sem chamadas externas de API) com uma [Referência de expressão JSonata](#) expressão para avaliar o agente do usuário.

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. Selecione Funções no painel de navegação.
3. Escolha a opção Criar função.
4. No modal do assistente de criação, selecione Criar do zero e escolha Continuar.
5. Em Tipo de função, selecione o quadro de saída personalizada.
6. Em Detalhes da função, insira o seguinte:
 - ID da função: myFirstFunction
 - Descrição: Classify device type from user agent
7. Em Configuração de saída personalizada, na seção Saída, adicione uma linha:
 - Chave: player_params.deviceType
 - Valor:

```
{% $contains(session.user_agent, 'CTV') ? 'ctv' :  
$contains(session.user_agent, 'Mobile') ? 'mobile' : 'desktop' %}
```
8. Escolha a opção Criar função.

Uma notificação de sucesso confirma que a função foi criada e você será redirecionado para a página de detalhes da função.

A configuração da função resultante é:

```
{
  "FunctionId": "myFirstFunction",
  "FunctionType": "CUSTOM_OUTPUT",
  "Description": "Classify device type from user agent",
  "CustomOutputConfiguration": {
    "Runtime": "JSONATA",
    "Output": {
      "player_params.deviceType": "{% $contains(session.user_agent, 'CTV') ?
'ctv' : $contains(session.user_agent, 'Mobile') ? 'mobile' : 'desktop' %}"
    }
  }
}
```

Etapa 2: anexar a função a uma configuração de reprodução

Mapeie a função para um gancho de ciclo de vida em sua configuração de reprodução. O mapeamento informa MediaTailor quando executar a função.

1. No painel de navegação, escolha Configurações.
2. Escolha a configuração de reprodução que você deseja atualizar.
3. Escolha Editar.
4. Expanda a seção Configuração de funções.
5. Para Gancho de inicialização da sessão, selecione no myFirstFunction menu suspenso.
6. Escolha Salvar.

Isso se conecta myFirstFunction ao gancho do [INICIALIZAÇÃO DA PRÉ-SESSÃO](#) ciclo de vida. O mapeamento de funções resultante é:

```
{
  "FunctionMapping": {
    "PRE_SESSION_INITIALIZATION": "myFirstFunction"
  }
}
```

MediaTailor executa a função uma vez no início de cada nova sessão nessa configuração de reprodução.

Etapa 3: iniciar uma sessão e verificar se a função foi executada

Inicie uma nova sessão de reprodução para acionar a função. Use uma solicitação de inicialização de sessão para o endpoint de inicialização de sessão da sua configuração de reprodução.

MediaTailor publica automaticamente CloudWatch métricas para cada execução de função, sem a necessidade de aceitação. Depois de iniciar uma sessão, verifique as seguintes métricas no AWS/MediaTailor namespace para confirmar se sua função foi executada:

- `PreSessionInitHook.Invocations`— Confirma que o gancho foi acionado.
- `PreSessionInitHook.Errors`— Deveria ser 0 se a função for bem-sucedida.
- `Function.Invocations`— Confirma a função individual executada. Essa métrica inclui `FunctionId`, `FunctionType`, e `HookType` dimensões para que você possa filtrar `myFirstFunction` especificamente.

Se a função falhar, MediaTailor emite eventos de registro de erros para os registros manifestos por padrão (sem necessidade de configuração):

- `PRE_SESSION_INIT_HOOK_ERROR`— Hook-level falha com `errorType` cause e.
- `PRE_SESSION_INIT_FUNCTION_ERROR`— Function-level falha com os detalhes específicos `functionId` e do erro.

O exemplo a seguir mostra um `PRE_SESSION_INIT_FUNCTION_ERROR` evento para um erro de sintaxe na expressão da função:

```
{
  "eventTimestamp": "2024-01-01T12:00:00.076000000Z",
  "eventType": "PRE_SESSION_INIT_FUNCTION_ERROR",
  "eventDescription": "Function execution failed",
  "awsAccountId": "123456789012",
  "originId": "my-config",
  "sessionId": "session-123",
  "requestId": "req-abc",
  "eventId": "5dc6f040-0f72-4e8c-a64e-25eeef62708c",
  "functionId": "myFirstFunction",
  "functionType": "CUSTOM_OUTPUT",
  "executionTimeMs": 2,
  "errorType": "SYNTAX_ERROR",
  "cause": "Expected \"\\\"\\\" before end of expression",
```



```
"input": {}  
}
```

Use o `eventId` campo para correlacionar eventos de erro de gancho e função para a mesma execução. O `errorType` campo indica a classe da falha — consulte [Solução de problemas e monitoramento](#) para obter uma lista completa dos tipos e correções de erros.

Note

Para obter um registro detalhado de sucesso, opte por participar `PRE_SESSION_INIT_HOOK_SUMMARY` e participar de `PRE_SESSION_INIT_FUNCTION_COMPLETED` eventos na configuração do Registro de Manifestos. Eventos resumidos mostram o resultado final de cada execução. Os eventos concluídos mostram a entrada, a saída e request/response os detalhes HTTP de cada função. Eles são desativados por padrão para minimizar os custos de registro. Para obter mais informações, consulte [Solução de problemas e monitoramento](#).

O que acontece nos bastidores

Aqui está o fluxo completo da solicitação para a função que você acabou de criar:

1. O jogador inicia uma sessão com MediaTailor.
2. MediaTailor dispara o gancho do `PRE_SESSION_INITIALIZATION` ciclo de vida e funciona. `myFirstFunction`
3. A função avalia o `session.user_agent` campo e grava `ctvmobile`, ou `desktop` `paraplayer_params.deviceType`.
4. MediaTailor cria a sessão e retorna o manifesto para o player.
5. O player encontra uma pausa publicitária durante a reprodução.
6. MediaTailor aciona o gancho do `PRE_ADS_REQUEST` ciclo de vida e, em seguida, constrói a solicitação ADS. Como `deviceType` está armazenado nos parâmetros do player, ele está disponível para inclusão no URL de solicitação do ADS por meio da substituição dinâmica de variáveis.
7. O ADS usa o tipo de dispositivo para retornar anúncios criativos direcionados.
8. MediaTailor junta os anúncios no manifesto e os devolve ao player.

Se a função falhar por algum motivo, MediaTailor descarta a saída e prossegue como se nenhuma função estivesse anexada. O espectador ainda vê anúncios, mas sem a segmentação por tipo de dispositivo.

Tópicos sugeridos

Agora você tem uma função funcional anexada a uma configuração de reprodução. A partir daqui:

- Para saber quais campos de entrada e namespaces de saída estão disponíveis em cada gancho do ciclo de vida, consulte [Hooks do ciclo de vida](#)
- Para saber mais sobre os diferentes tipos de função e como encadeá-los, consulte [Tipos de funções e composição](#).
- Para ver exemplos completos de trabalho, consulte [Exemplos de funções](#).

Ganchos de ciclo de vida de funções

Um gancho de ciclo de vida define quando sua função MediaTailor é executada durante a reprodução. Esta página é uma referência completa para campos de entrada, namespaces de saída e as regras que governam o fluxo de dados em cada gancho.

Visão geral do

MediaTailor suporta dois ganchos de ciclo de vida:

- **PRE_SESSION_INITIALIZATION** é acionado uma vez quando um espectador inicia uma nova sessão. Use-o para um trabalho de configuração único, como buscar segmentos de público. Neste momento, nenhum intervalo de anúncio ocorreu, portanto, o contexto do intervalo de anúncio não está disponível.
- **PRE_ADS_REQUEST** é acionado antes de cada solicitação do servidor de decisão de anúncio (ADS) — uma vez por intervalo de anúncio na transmissão. Use-o para personalizar a solicitação do ADS com dados de segmentação, modificar o URL do ADS ou adicionar cabeçalhos.

A principal diferença é o tempo: é PRE_SESSION_INITIALIZATION executado uma vez e configura dados que persistem durante toda a sessão, ao mesmo tempo em que PRE_ADS_REQUEST é executado repetidamente e pode adaptar cada solicitação do ADS ao intervalo de anúncio específico.

INICIALIZAÇÃO DA PRÉ-SESSÃO

Quando dispara

MediaTailor executa a função mapeada para `PRE_SESSION_INITIALIZATION` uma vez, no início de uma nova sessão de reprodução. A função é executada antes de MediaTailor construir a resposta inicial do manifesto.

Input

`session.*`, `player_params.*` e `event.*`. Para ver todos os campos disponíveis, consulte [Referência do campo de entrada](#).

Namespace de saída permitido

Namespace	Tipos aceitos
<code>player_params.*</code>	Cadeias de caracteres, números, booleanos

Os valores `player_params.*` gravados são mantidos na sessão. Eles estão disponíveis:

- Como entrada no `PRE_ADS_REQUEST` ciclo de vida, gancho via `player_params.*`
- No ADS, solicite URLs por meio de [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#) (por exemplo, `[player_params.deviceType]`)
- Durante toda a duração da sessão em todos os intervalos publicitários

Note

O tamanho total serializado de todas as chaves e valores de `player_params` saída não deve exceder 1.000 caracteres. Se o total exceder esse limite, a saída da função será descartada. Para obter mais informações, consulte [Limites de funções](#).

Caso de uso típico

- Obtenha dados de identidade ou público de um serviço externo e armazene-os nos parâmetros do player para uso em solicitações posteriores do ADS.

- Classifique o tipo de dispositivo com base no agente do usuário e grave a classificação em um parâmetro do player.
- Defina os valores padrão dos parâmetros do player nos quais o processamento posterior do anúncio depende.
- Armazene valores nos parâmetros do player que estão incluídos no URL do ADS por meio de [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#).

Comportamento com falha

Se uma função anexada `PRE_SESSION_INITIALIZATION` falhar por qualquer motivo, MediaTailor descarta a saída da função e procede como se nenhuma função estivesse anexada. A sessão começa normalmente sem os valores dos parâmetros do player da função.

SOLICITAÇÃO PRÉVIA DE ANÚNCIOS

Quando dispara

MediaTailor executa a função mapeada `PRE_ADS_REQUEST` uma vez por intervalo de anúncio, imediatamente antes de enviar a solicitação ao ADS. A função é executada sempre que uma oportunidade de anúncio é encontrada durante o processamento do manifesto.

Input

Todos os campos de `PRE_SESSION_INITIALIZATION`, mais `avail.*`, `scte.*`, `asset.*`, e `adsRequest.*` (url, método, cabeçalhos, corpo). Para ver todos os campos disponíveis, consulte [Referência do campo de entrada](#).

Namespace de saída permitido

Namespace	Tipos aceitos	Como a saída é usada
<code>player_params.*</code>	Cadeias de caracteres, números, booleanos	Substitui os parâmetros do player da sessão para esse intervalo publicitário. Disponível para o URL de solicitação do ADS por meio de MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS .

Namespace	Tipos aceitos	Como a saída é usada
<code>session.*</code>	Cadeias de caracteres, números, booleanos	Substitui as variáveis de sessão desse intervalo publicitário. Disponível para o URL de solicitação do ADS por meio da substituição dinâmica de variáveis.
<code>avail.*</code>	Cadeias de caracteres, números, booleanos	Substitui as variáveis disponíveis para esse intervalo publicitário. Disponível para o URL de solicitação do ADS por meio da substituição dinâmica de variáveis.
<code>scte.*</code>	Cadeias de caracteres, números, booleanos	Substitui as variáveis SCTE para esse intervalo de anúncio. Disponível para o URL de solicitação do ADS por meio da substituição dinâmica de variáveis.
<code>asset.*</code>	Cadeias de caracteres, números, booleanos	Substitui as variáveis de metadados do ativo (das EXT-X-ASSET tags) desse intervalo publicitário. Disponível para o URL de solicitação do ADS por meio da substituição dinâmica de variáveis. Para obter mais informações sobre marcadores de anúncios HLS, consulte Marcadores de anúncios HLS .

Namespace	Tipos aceitos	Como a saída é usada
<code>adsRequest.*</code>	String	Substitui a solicitação do ADS somente para esse intervalo publicitário. Campos suportados: <code>url,method,headers.<name>,body</code> . O <code>url</code> valor é tratado como um modelo e é suportado MediaTailor ou variáveis dinâmicas de anúncios para solicitações de ADS após a execução da função. Não persistiu.

Note

Todas as saídas do `PRE_ADS_REQUEST` gancho são substituições transitórias — elas se aplicam somente à solicitação de ADS do intervalo publicitário atual e não persistem na sessão.

Exemplo — reescrever a solicitação do ADS:

```
{
  "Output": {
    "adsRequest.url": "{%'https://ads.example.com/v1/vast?sid=' & session.id &
'&genre=' & player_params.genre%}",
    "adsRequest.headers.X-Custom-Token": "{%player_params.auth_token%}"
  }
}
```

Isso substitui o URL do ADS e adiciona um cabeçalho personalizado para o intervalo publicitário atual.

Caso de uso típico

- Reescreva o URL de solicitação do ADS para rotear o tráfego entre diferentes servidores de anúncios para A/B testes.
- Anexe dados de enriquecimento (segmentos de público, tokens de identidade) ao URL ou aos cabeçalhos da solicitação do ADS.
- Modifique condicionalmente os parâmetros de solicitação do ADS com base nos dados SCTE-35 do sinal ou no índice de disponibilidade.

Comportamento com falha

Se uma função anexada PRE_ADS_REQUEST falhar por qualquer motivo, MediaTailor descarta a saída da função e procede como se nenhuma função estivesse anexada. A solicitação do ADS é enviada usando os parâmetros originais da sessão e da solicitação sem modificação.

Referência do campo de entrada

Campo	Tipo	INICIALIZAÇÃO DA PRÉ-SESSÃO	SOLICITAÇÃO PRÉVIA DE ANÚNCIOS
session.id	Longo	✓	✓
session.uuid	String	✓	✓
session.client_ip	String	✓	✓
session.user_agent	String	✓	✓
session.referer *	String	✓	✓
session.available_duration_secs	Longo	✗	✓

Campo	Tipo	INICIALIZAÇÃO DA PRÉ-SESSÃO	SOLICITAÇÃO PRÉVIA DE ANÚNCIOS
session.avail_duration_ms	Longo	✗	✓
player_params.*	String	✓	✓
event.id	String	✓	✓
event.hook	String	✓	✓
event.timestamp	String	✓	✓
avail.index	Int	✗	✓
avail.random	Longo	✗	✓
avail.source_content_time_epoch_ms	Longo	✗	✓
scte.event_id	Int	✗	✓
scte.avail_num	Int	✗	✓
scte.segmentation_event_id	Int	✗	✓
scte.segmentation_type_id	Int	✗	✓
scte.segmentation_upid	String	✗	✓

Campo	Tipo	INICIALIZAÇÃO DA PRÉ-SESSÃO	SOLICITAÇÃO PRÉVIA DE ANÚNCIOS
scte.segm entation_ upid.assetId	String	x	✓
scte.segm entation_ upid.cueD ata.key	String	x	✓
scte.segm entation_ upid.cueD ata.value	String	x	✓
scte.uniq ue_program_id	Int	x	✓
scte.arch ive_allow ed_flag	Booleano	x	✓
scte.deliv ery_not_ restricte d_flag	Booleano	x	✓
scte.devi ce_restri ctions	Int	x	✓
scte.no_r egional_b lackout_flag	Booleano	x	✓

Campo	Tipo	INICIALIZAÇÃO DA PRÉ-SESSÃO	SOLICITAÇÃO PRÉVIA DE ANÚNCIOS
<code>scte.segment_num</code>	Int	X	✓
<code>scte.segment_ents_expected</code>	Int	X	✓
<code>scte.sub_segment_num</code>	Int	X	✓
<code>scte.sub_segments_expected</code>	Int	X	✓
<code>scte.avails_expected</code>	Longo	X	✓
<code>asset.*</code>	String	X	✓
<code>adsRequest.url</code>	String	X	✓
<code>adsRequest.method</code>	String	X	✓
<code>adsRequest.headers.<key></code>	String	X	✓
<code>adsRequest.body</code>	String	X	✓

* só `session.referer` está presente quando um cabeçalho Referer é incluído na solicitação de inicialização da sessão. Use `$exists(session.referer)` para verificar antes de acessar.

Tipos de funções e composição

AWS Elemental MediaTailor fornece três tipos de função, cada um projetado para uma categoria diferente de trabalho. Você escolhe um tipo ao criar uma função, e o tipo determina o que a função pode fazer em tempo de execução. Esta página explica como cada tipo é executado, quando usá-lo e como compor funções em pipelines de várias etapas.

Visão geral do tipo de função

Tipo	Categoria	Finalidade
CUSTOM_OUTPUT	Transformação de dados	Avalie as expressões em relação ao estado atual da sessão e produza saídas. Sem chamadas externas.
HTTP_REQUEST	Integração externa	Faça uma chamada HTTP para um serviço externo e, em seguida, avalie as expressões de saída que possam referenciar a resposta.
SEQUENTIAL_EXECUTOR	Orquestração	Execute uma sequência de funções em ordem, passando dados entre etapas por meio de dados temporários.

Cada tipo tem uma função distinta. Escolha um tipo ao criar uma função — o tipo determina o que a função pode fazer em tempo de execução.

SAÍDA_PERSONALIZADA

Quando usar

Use CUSTOM_OUTPUT quando sua lógica de função for puramente computacional. Os casos de uso comuns incluem a classificação de dispositivos com base nas cadeias de caracteres do agente do usuário, a seleção de URLs do servidor de decisão de anúncios com a lógica de A/B teste e a reformatação dos parâmetros do player antes que eles cheguem ao servidor de decisão do anúncio.

CUSTOM_OUTPUTAs funções são do tipo mais rápido porque avaliam expressões na memória sem chamadas de rede. Escolha esse tipo sempre que puder atingir sua meta sem entrar em contato com um serviço externo.

Campos de configuração

Uma CUSTOM_OUTPUT função tem dois campos:

- Runtime — A linguagem de expressão. Defina isso comoJSONATA.
- Saída — Um mapa de pares de valores-chave. Cada chave é um nome de associação de saída (como`player_params.device_type`) e cada valor é uma expressão que é MediaTailor avaliada em tempo de execução.

Note

Use `{%...%}` para agrupar expressões JSonata em campos de string. Valores sem esses delimitadores são tratados como cadeias de caracteres estáticas. Por exemplo, `"{%session.id%}"` avalia a expressão, enquanto `"session.id"` é o texto literal.

Comportamento durante a execução

Quando MediaTailor executa uma CUSTOM_OUTPUT função, ela avalia cada expressão no bloco de saída em relação ao estado atual da sessão. Todas as expressões veem o mesmo instantâneo de entrada. Os resultados avaliados se tornam as saídas da função.

Se alguma expressão falhar na avaliação, MediaTailor descarta toda a saída da função e prossegue como se nenhuma função estivesse anexada. A função não confirma parcialmente os resultados.

Exemplo: divisão A/B de tráfego

A função a seguir atribui aleatoriamente cada intervalo de anúncio a um dos dois URLs do servidor de decisão de anúncios. Ele foi projetado para o gancho do PRE_ADS_REQUEST ciclo de vida.

```
{
  "FunctionId": "abTestAdsUrl",
  "FunctionType": "CUSTOM_OUTPUT",
  "CustomOutputConfiguration": {
    "Runtime": "JSONATA",
```

```
"Output": {
  "adsRequest.url": "{%$random() < 0.5 ? 'https://ads.example.com/v1' :
'https://ads.example.com/v2'%}"
}
```

Para obter uma explicação completa desse exemplo, consulte [Exemplos de funções](#)

HTTP_REQUEST

Quando usar

Use HTTP_REQUEST quando sua função precisar chamar um serviço externo. Os casos de uso comuns incluem a busca de dados de identidade de um provedor de resolução, a recuperação de segmentos de público de uma plataforma de gerenciamento de dados e o envio de informações da sessão para um endpoint de registro.

Campos de configuração

Uma HTTP_REQUEST função tem os seguintes campos:

- Runtime — A linguagem de expressão. Defina isso comoJSONATA.
- MethodType— O método HTTP. Os valores compatíveis são GET e POST.
- Url — O URL para o qual enviar a solicitação. Você pode usar uma URL estática ou uma expressão JSONATA que cria a URL dinamicamente.
- Cabeçalhos — Os cabeçalhos HTTP a serem incluídos na solicitação, especificados como pares de nome e valor do cabeçalho. Use a sintaxe de {%...%} expressão para valores de cabeçalho dinâmicos. Valores estáticos podem ser especificados diretamente como cadeias de caracteres.
- Corpo — O corpo da solicitação a ser enviada. Usado com POST solicitações. Você pode usar uma expressão JSonata para criar o corpo dinamicamente.
- RequestTimeoutMilliseconds(obrigatório) — Quanto tempo esperar por uma resposta.
- Saída — Define os valores a serem produzidos após a conclusão da chamada HTTP. Cada entrada mapeia uma chave de saída (comoplayer_params.envelope_id) para uma expressão que pode referenciar o response objeto.

Para ver os limites e restrições de tamanho que se aplicam a esses campos, consulte[Limites](#).

Como a solicitação é processada

MediaTailor processa uma `HTTP_REQUEST` função em duas etapas:

1. Crie a solicitação — MediaTailor avalia as `Body` expressões `Url`, `Headers`, e em relação ao estado atual da sessão. Esses valores avaliados formam a solicitação HTTP de saída.
2. Processe a resposta — Após a conclusão da chamada HTTP, MediaTailor avalia as expressões no bloco de saída. Essas expressões podem fazer referência tanto ao estado original da sessão quanto ao `response` objeto retornado pela chamada.

Campos de resposta

Depois que a chamada HTTP for concluída, você poderá referenciar os seguintes campos em suas expressões de saída:

Campo	Tipo	Description
<code>response.body</code>	Objeto ou matriz	O corpo da resposta foi analisado como JSON. Defina como <code>null</code> se o corpo exceder 20.000 caracteres ou não for um JSON válido.
<code>response.statusCode</code>	Inteiro	O código de status HTTP retornado pelo serviço externo. Defina como <code>null</code> em caso de falha de rede.
<code>response.text</code>	String	O corpo bruto da resposta como uma string, truncado para 20.000 caracteres. Defina como <code>"Internal Error"</code> em caso de falha de rede.

 Important

O `response.body` campo é `null` quando a resposta excede 20.000 caracteres, mesmo que a resposta seja um JSON válido.

 Note

O objeto de resposta está disponível somente no bloco de saída de uma `HTTP_REQUEST` função. Você não pode referenciar campos de resposta nos campos URL, cabeçalhos ou corpo. Em `aSEQUENTIAL_EXECUTOR`, cada `HTTP_REQUEST` função pode acessar somente sua própria resposta.

Um valor de `null` significa que os dados não estão disponíveis. Isso acontece quando a chamada HTTP falha (erro de rede ou tempo limite) ou quando o corpo da resposta excede 20.000 caracteres ou não é um JSON válido.

Comportamento de falha de rede

Se a chamada HTTP falhar devido a um erro de rede ou tempo limite `response.statusCode` e `response.body` estiver definida como e `response.text` definida como `null` "Internal Error" Suas expressões de saída ainda são executadas, portanto, sempre verifique `response.statusCode` antes de usar os dados de resposta.

 Tip

Use uma expressão condicional para lidar com falhas normalmente:
`{%response.statusCode = 200 ? response.body.value : 'default%'}`

Exemplo: buscar dados de identidade

A função a seguir chama uma API de resolução de identidade no início da sessão e armazena o resultado nos parâmetros do player. Ele foi projetado para o gancho do `PRE_SESSION_INITIALIZATION` ciclo de vida.

```
{
```

```

    "FunctionId": "fetchIdentityEnvelope",
    "FunctionType": "HTTP_REQUEST",
    "HttpRequestConfiguration": {
        "Runtime": "JSONATA",
        "MethodType": "GET",
        "Url": "{%'https://identity.example.com/v1/resolve?ip=' &
$encodeURIComponent(session.client_ip)%}",
        "Headers": {
            "Authorization": "{%'Bearer my_api_token'%}",
            "Accept": "application/json"
        },
        "RequestTimeoutMilliseconds": 2000,
        "Output": {
            "player_params.identity_envelope": "{%response.statusCode = 200 ?
response.body.envelope : ''%}"
        }
    }
}

```

Para obter uma explicação completa de um exemplo semelhante, consulte [Exemplos de funções](#)

EXECUTOR_SEQUENCIAL

Quando usar

A `SEQUENTIAL_EXECUTOR` executa uma série de funções, uma de cada vez, em ordem. Cada etapa pode usar os resultados das etapas anteriores.

Use `SEQUENTIAL_EXECUTOR` quando sua lógica exigir várias etapas que dependem dos resultados umas das outras. Os casos de uso comuns incluem buscar dados de identidade e usá-los para recuperar segmentos de público, executar uma etapa de classificação e, em seguida, chamar condicionalmente diferentes serviços externos com base no resultado e criar URLs complexos de solicitações de anúncios a partir de várias fontes de dados.

Campos de configuração

Uma `SEQUENTIAL_EXECUTOR` função tem os seguintes campos:

- **Runtime** — A linguagem de expressão. Defina isso como `JSONATA`.
- **FunctionList** — Uma lista ordenada de 1 a 10 etapas. Cada etapa especifica a função `FunctionId` a ser executada. Opcionalmente, você pode adicionar uma `RunCondition` expressão para controlar se a etapa é executada ou ignorada.

- **Saída** — Define os valores a serem produzidos após a conclusão de todas as etapas. Cada entrada mapeia uma chave de saída (`comoplayer_params.envelope`) para uma expressão que pode referenciar dados produzidos por qualquer etapa na sequência. Se for omitida, toda a saída das funções individuais na sequência será usada.
- **TimeoutMilliseconds(obrigatório)** — O tempo máximo para a conclusão de toda a sequência. Se a sequência exceder esse tempo limite, MediaTailor descartará toda a saída da sequência.

Execução ordenada e fluxo de dados

MediaTailor executa cada etapa da sequência da primeira à última. Após a conclusão de cada etapa, os valores que ela produz são mesclados em um conjunto contínuo de resultados. As etapas subsequentes podem acessar os dados da sessão original mais todos os valores produzidos pelas etapas anteriores.

Os dados temporários são o principal mecanismo para transmitir dados entre as etapas. Quando uma função grava em uma `temp.*` chave, a próxima etapa pode ler esse valor. Os parâmetros do player e os campos de solicitação de anúncios escritos nas etapas anteriores também são visíveis nas etapas posteriores.

Note

Os dados temporários aceitam qualquer tipo de dados, incluindo objetos e matrizes. Os parâmetros do player e os campos de solicitação de anúncio aceitam somente sequências de caracteres, números, booleanos e nulos).

Per-step condições de execução

Cada etapa da sequência tem um `RunCondition` campo opcional. Esse campo contém uma expressão que retorna `true` ou `false`. MediaTailor avalia a `RunCondition` expressão imediatamente antes de executar essa etapa.

Se a `RunCondition` expressão for avaliada como `false`, MediaTailor pulará totalmente a etapa e passará para a próxima. Se o `RunCondition` campo for omitido, a etapa sempre será executada.

```
{ "FunctionId": "retryFetch", "RunCondition": "{%temp.statusCode = 500%}" }
```

Esse mecanismo permite criar pipelines condicionais. Por exemplo, você pode executar uma busca de identidade na etapa 1 e, em seguida, executar condicionalmente uma pesquisa de segmento na etapa 2 somente se a etapa 1 retornar uma identidade válida.

Como funciona o bloco de saída

O bloco de saída em um `SEQUENTIAL_EXECUTOR` controla o que a sequência produz após a conclusão de todas as etapas:

- Bloco de saída presente — MediaTailor avalia as expressões no bloco de saída em relação ao estado acumulado final e salva somente essas saídas. Todas as saídas produzidas por etapas anteriores que não são referenciadas no bloco de saída sequencial são descartadas.
- Bloco de saída ausente — MediaTailor salva diretamente todas as saídas acumuladas de todas as etapas.

 Tip

Omita o bloco de saída quando quiser que a saída de cada função passe. Adicione um bloco de saída quando precisar filtrar, renomear ou transformar os resultados acumulados antes de salvá-los.

Configuração de tempo limite

O `TimeoutMilliseconds` campo define um prazo para toda a sequência. Esse tempo limite abrange todas as etapas, incluindo todas as chamadas HTTP feitas por funções. Se a sequência exceder o tempo limite, MediaTailor descartará toda a saída da sequência e prosseguirá como se nenhuma função estivesse anexada.

HTTP_REQUESTAs funções individuais ainda respeitam sua própria `RequestTimeoutMilliseconds` configuração. O tempo limite da sequência atua como um limite externo que limita o tempo total de execução.

Exemplo: tente novamente em caso de falha de HTTP

Este exemplo chama uma API de identidade e tenta novamente automaticamente se a primeira chamada retornar um erro no servidor. Ele usa duas funções `HTTP_REQUEST` orquestradas por um `SEQUENTIAL_EXECUTOR`.

Etapa 1 — Busca primária (**`fetchIdentity`**):

```
{
  "FunctionId": "fetchIdentity",
  "FunctionType": "HTTP_REQUEST",
  "HttpRequestConfiguration": {
    "Runtime": "JSONATA",
    "MethodType": "GET",
    "Url": "{%'https://identity.example.com/v1/resolve?ip=' & session.client_ip%}",
    "Headers": {
      "Accept": "application/json"
    },
    "RequestTimeoutMilliseconds": 1000,
    "Output": {
      "temp.statusCode": "{%response.statusCode%}",
      "temp.envelope": "{%response.statusCode = 200 ? response.body.envelope :
null%}"
    }
  }
}
```

Etapa 2 — Tente novamente em caso de falha (**retryIdentity**):

```
{
  "FunctionId": "retryIdentity",
  "FunctionType": "HTTP_REQUEST",
  "HttpRequestConfiguration": {
    "Runtime": "JSONATA",
    "MethodType": "GET",
    "Url": "{%'https://identity-fallback.example.com/v1/resolve?ip=' &
session.client_ip%}",
    "Headers": {
      "Accept": "application/json"
    },
    "RequestTimeoutMilliseconds": 1000,
    "Output": {
      "temp.statusCode": "{%response.statusCode%}",
      "temp.envelope": "{%response.statusCode = 200 ? response.body.envelope :
null%}"
    }
  }
}
```

Sequência (**identityWithRetry**):

```
{
  "FunctionId": "identityWithRetry",
  "FunctionType": "SEQUENTIAL_EXECUTOR",
  "SequentialExecutorConfiguration": {
    "Runtime": "JSONATA",
    "TimeoutMilliseconds": 2000,
    "FunctionList": [
      { "FunctionId": "fetchIdentity" },
      { "FunctionId": "retryIdentity", "RunCondition": "{%temp.statusCode >= 500%}" }
    ],
    "Output": {
      "player_params.envelope": "{%temp.envelope%}"
    }
  }
}
```

Como funciona:

1. `fetchIdentity` chama a API de identidade e grava o código de status e o envelope em `temp.*`.
2. Se o código de status for 500 ou superior, a `RunCondition` etapa 2 será avaliada `true` e `retryIdentity` executada. Ele sobrescreve `temp.statusCode` e `temp.envelope` com a resposta de nova tentativa.
3. Se a primeira chamada for bem-sucedida, a etapa 2 será ignorada.
4. O bloco de saída da sequência grava `temp.envelope` em `player_params.envelope`.

Regras de composição

MediaTailor impõe os seguintes limites na composição da função:

Regra	Limite
Profundidade máxima de nidificação	2 (uma sequência pode conter funções, mas essas crianças não podem ser sequências em si mesmas)
Etapas por sequência	1 a 10

Regra	Limite
Total de execuções de funções por gancho do ciclo de vida	20
Referências circulares	Não é permitido. Uma função não pode referenciar a si mesma, direta ou indiretamente.
Existência da função	Todas as funções referenciadas em a <code>FunctionList</code> devem existir antes de você criar ou atualizar a sequência principal.

Quando você cria uma função, MediaTailor valida a sintaxe da expressão, verifica se há funções restritas, verifica se todas as funções referenciadas existem e detecta referências circulares. Quando você anexa uma função a uma configuração de reprodução, valida MediaTailor adicionalmente se todas as chaves de saída em toda a árvore de funções são compatíveis com o gancho do ciclo de vida atribuído.

Criação e gerenciamento de funções

Esta página explica como criar, anexar, atualizar e excluir funções usando o MediaTailor console. As funções permitem que você personalize o comportamento da sessão e as solicitações de anúncios executando a lógica em pontos-chave durante a reprodução.

Configurando uma função

Criando uma função usando o console

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. Selecione Funções no painel de navegação.
3. Escolha a opção Criar função.
4. Em ID da função, insira um nome exclusivo para a função (por exemplo, `fetchIdentity`).
5. Em Tipo de função `CUSTOM_OUTPUT`, escolha `HTTP_REQUEST`, ou `SEQUENTIAL_EXECUTOR`. O console exibe os campos de configuração do tipo selecionado.
6. Preencha os campos de configuração específicos do tipo. Para obter uma descrição de cada tipo de função e seus campos, consulte [Tipos de funções e composição](#).

7. Escolha a opção Criar função.

Anexando uma função a uma configuração de reprodução

Uma função não é executada até que você a anexe a uma configuração de reprodução por meio de um mapeamento de função (um link entre um gancho de ciclo de vida e uma função). O mapeamento da função especifica qual gancho do ciclo de vida aciona a função.

1. Abra o MediaTailor console.
2. No painel de navegação, escolha Configurações.
3. Escolha a configuração de reprodução que você deseja atualizar.
4. Na seção Mapeamento de funções, escolha Editar.
5. Para cada gancho de ciclo de vida, selecione a função a ser conectada:
 - Gancho de inicialização da sessão — Escolha uma função para executar uma vez no início da sessão.
 - Gancho de solicitação de anúncio — Escolha uma função a ser executada antes de cada solicitação do ADS.
6. Escolha Salvar.

Gerenciar funções

Funções de visualização

Navegue até Funções para ver todas as funções em sua conta. Escolha um nome de função para ver sua configuração.

Atualizando uma função existente

Para atualizar uma função, navegue até Funções, escolha a função e modifique a configuração. A atualização substitui toda a definição da função.

Tip

Salve uma cópia da configuração da função antes de fazer alterações. Não há controle de versão ou reversão embutido para funções.

Removendo uma função

Separando uma função de uma configuração de reprodução

Antes de excluir uma função, remova-a de todas as configurações de reprodução que fazem referência a ela.

1. Abra o MediaTailor console.
2. No painel de navegação, escolha Configurações.
3. Escolha a configuração de reprodução.
4. Na seção Mapeamento de funções, escolha Editar.
5. Remova a função do gancho do ciclo de vida.
6. Escolha Salvar.

Como excluir uma função

Navegue até Funções, selecione a função e escolha Excluir.

Regras de bloqueio de exclusões

MediaTailor impede que você exclua uma função que ainda está em uso.

Condição	Resultado
A função é anexada a uma configuração de reprodução por meio de mapeamento de funções	A exclusão está bloqueada. Desconecte a função primeiro.
A função é referenciada em um SEQUENTIAL_EXECUTOR	A exclusão está bloqueada. Remova primeiro a referência da função principal.
A função não é referenciada em nenhum lugar	A exclusão foi bem-sucedida.

Regras de validação

MediaTailor valida sua função ao criá-la ou atualizá-la. As seguintes verificações são realizadas:

- Sintaxe da expressão — Todas as expressões devem ser JSONATA válidas.

- Funções restritas — As expressões não podem chamar funções restritas do JSonata. Veja [Referência de expressão JSonata](#) a lista completa.
- Prefixos de chave de saída — Todas as chaves de saída devem começar com um prefixo de namespace reconhecido. Para obter a lista de prefixos aceitos, consulte [Hooks do ciclo de vida](#).
- Referências de função — Todos os IDs de função em uma FunctionList devem fazer referência a funções existentes.
- Referências circulares — Uma função não pode referenciar a si mesma, direta ou indiretamente.
- Profundidade de aninhamento — A SEQUENTIAL_EXECUTOR pode chamar outras funções, mas essas funções não podem ser em si mesmas SEQUENTIAL_EXECUTOR.

Para valores específicos e limites de tamanho, consulte [Limites](#).

Referência de API

Para gerenciar funções de forma programática, consulte a Referência da [AWS Elemental MediaTailor API](#).

Referência de expressão JSonata para Funções

Esta página é uma referência completa para a sintaxe da expressão, operadores e funções disponíveis em Funções. Use-o ao escrever expressões para blocos de saída, campos de URL, valores de cabeçalho, modelos de corpo e condições de execução.

Delimitadores de expressão

Cada valor que você define em uma função é uma constante ou uma expressão, não uma mistura dos dois. MediaTailor distingue entre os dois com base em delimitadores.

Sintaxe	Tipo	Avaliação
<code>https://ads.example.com/vast</code>	Constante	Retornado no estado em que se encontra, sem avaliação.
<code>{%session.client_ip%}</code>	Expressão	Avaliado em tempo de execução. O resultado substitui o valor inteiro.

Sintaxe	Tipo	Avaliação
GET	Constante	Devolvido como está.
{%'https://ads.example.com/vast?ip=' & session.client_ip%}	Expressão	Avaliado em tempo de execução.

Important

Um valor é inteiramente uma constante ou inteiramente uma expressão. Você não pode misturar os dois em um único valor. Por exemplo, `hello {%'world'%}` não é válido. Para combinar texto estático com valores dinâmicos, use a concatenação de strings dentro da expressão: `{%'hello ' & 'world'%}`

Fundamentos da linguagem

Notação de pontos para navegação por caminhos

Use a notação de pontos para percorrer os dados de entrada. Cada ponto desce um nível na hierarquia do objeto.

```
session.client_ip           # the viewer's IP address
response.body.envelope      # a field inside a parsed JSON response
player_params.campaign_id   # a player parameter
```

Os campos ausentes retornam `null` sem gerar um erro:

```
temp.nonExistent           # null
temp.nonExistent.deeply.nested # null
```

Concatenação de strings com **&**

O **&** operador une dois valores de cadeia de caracteres. Non-string os valores são convertidos em cadeias de caracteres automaticamente.

```
'https://ads.example.com/vast?ip=' & session.client_ip
```

```
# "https://ads.example.com/vast?ip=192.0.2.1"

'duration=' & 30
# "duration=30"
```

Expressões condicionais (ternárias)

Use o operador ternário para retornar um dos dois valores com base em uma condição.

```
condition ? value_if_true : value_if_false
```

Exemplos:

```
$exists(player_params.env) ? player_params.env : 'prod'
response.statusCode = 200 ? response.body.id : 'unknown'
$random() > 0.5 ? 'groupA' : 'groupB'
```

Você pode aninhar expressões ternárias para ramificação multidirecional:

```
$contains(session.user_agent, 'CTV') ? 'ctv'
  : $contains(session.user_agent, 'Mobile') ? 'mobile'
  : 'desktop'
```

Vinculação variável com: =

Use o := operador entre parênteses para atribuir valores intermediários em uma expressão. As variáveis vinculadas têm como escopo os parênteses delimitados e não persistem fora da expressão.

```
(
  $base := 'https://ads.example.com';
  $base & '/vast?ip=' & session.client_ip
)
```

O ponto e vírgula separa as declarações entre parênteses. A última declaração é o valor de retorno da expressão.

```
(
  $code := response.statusCode;
  $code != null and $code >= 200 and $code < 300
)
```

```
? response.body.value  
: 'fallback'  
)
```

Operadores

Aritmética

Operador	Description	Exemplo	Resultado
+	Adição	5 + 3	8
-	Subtração	10 - 4	6
*	Multiplicação	6 * 7	42
/	Divisão	15 / 4	3.75
%	Módulo	17 % 5	2

Important

Os valores de entrada dos parâmetros do player e dos dados da sessão chegam como strings. Use `$number()` para convertê-los antes de comparações numéricas ou aritméticas. A comparação de uma string com um número produz resultados inesperados.

Comparação

Operador	Description	Exemplo	Resultado
=	Igual a	<code>response. statusCode = 200</code>	true
!=	Não é igual a	<code>player_pa rams.region != 'us-east-1'</code>	true se não us-east-1

Operador	Description	Exemplo	Resultado
<	Menor que	<code>avail.index < 3</code>	true se abaixo de 3
>	Maior que	<code>\$number(player_params.age) > 18</code>	true se tiver mais de 18 anos
<=	Menor ou igual a	<code>\$count(items) <= 10</code>	true se 10 ou menos
>=	Maior ou igual a	<code>response.statusCode >= 400</code>	true se o status do erro

Booleano

Operador	Description	Exemplo
and	AND lógico	<code>response.statusCode = 200 and \$exists(response.body.id)</code>
or	OR lógico	<code>player_params.region = 'us-east-1' or player_params.region = 'us-west-2'</code>

Use parênteses para precedência:

```
score > 0.5 and (tier = 'premium' or tier = 'gold')
```

Note

Use `$not()` para negação lógica. Não há operador de not palavra-chave.

Filiação (em)

O `in` operador testa se existe um valor em uma matriz.

```
'premium' in segments      # true if segments contains 'premium'
player_params.region in ['us-east-1', 'us-west-2'] # true
```

Encadeamento (~) >

O operador de encadeamento passa o resultado da expressão à esquerda como o primeiro argumento para a função à direita.

```
session.user_agent ~> $lowercase ~> $trim
# equivalent to $trim($lowercase(session.user_agent))
```

Funções permitidas

MediaTailor suporta as seguintes funções integradas. Qualquer função não listada aqui é bloqueada e causa um erro de validação quando você cria ou atualiza uma função.

Conversão de tipo (3)

Função	Description	Exemplo	Resultado
<code>\$string(value)</code>	Converter em string	<code>\$string(200)</code>	"200"
<code>\$number(value)</code>	Converter em número	<code>\$number('42')</code>	42
<code>\$boolean(value)</code>	Converter em booleano	<code>\$boolean(1)</code>	true

Introspecção (4)

Função	Description	Exemplo	Resultado
<code>\$length(string)</code>	Tamanho da segmento	<code>\$length('hello')</code>	5

Função	Description	Exemplo	Resultado
<code>\$count(array)</code>	Contagem de elementos da matriz	<code>\$count([1, 2, 3])</code>	3
<code>\$exists(value)</code>	Verifique se o valor existe (não indefinido)	<code>\$exists(emp.id)</code>	true ou false
<code>\$keys(object)</code>	Obter nomes de chaves de objetos	<code>\$keys(response.body)</code>	<code>["id", "name"]</code>

Numérico (7)

Função	Description	Exemplo	Resultado
<code>\$sum(array)</code>	Soma da matriz	<code>\$sum([1, 2, 3])</code>	6
<code>\$max(array)</code>	Valor máximo	<code>\$max([10, 5, 20])</code>	20
<code>\$min(array)</code>	Valor mínimo	<code>\$min([10, 5, 20])</code>	5
<code>\$average(array)</code>	Média aritmética	<code>\$average([10, 20, 30])</code>	20
<code>\$abs(number)</code>	Valor absoluto	<code>\$abs(-7)</code>	7
<code>\$floor(number)</code>	Arredondar para baixo	<code>\$floor(3.9)</code>	3
<code>\$round(number, precision)</code>	Redondo até a precisão	<code>\$round(3.456, 2)</code>	3.46

Corda (7)

Função	Description	Exemplo	Resultado
<code>\$uppercase(string)</code>	Para maiúsculas	<code>\$uppercase('hello')</code>	"HELLO"
<code>\$lowercase(string)</code>	Para minúsculas	<code>\$lowercase('Hello')</code>	"hello"
<code>\$trim(string)</code>	Remover espaço em leading/trailing branco	<code>\$trim(' hi ')</code>	"hi"
<code>\$substring(string, start, length)</code>	Extraia a substring (com base em zero)	<code>\$substring('abcdef', 2, 3)</code>	"cde"
<code>\$contains(string, pattern)</code>	Verifique se a string contém um padrão	<code>\$contains(session.user_agent, 'CTV')</code>	true ou false
<code>\$match(string, pattern)</code>	Combine a string com o padrão regex	<code>\$match('abc-123', /[0-9]+/)</code>	{"match": "123", ...}
<code>\$replace(string, pattern, replacement)</code>	Substitua o padrão correspondente	<code>\$replace('hello', 'l', 'r')</code>	"herro"

Matriz (5)

Função	Description	Exemplo	Resultado
<code>\$append(arr1, arr2)</code>	Concatenar matrizes	<code>\$append([1, 2], [3, 4])</code>	[1, 2, 3, 4]

Função	Description	Exemplo	Resultado
<code>\$reverse(array)</code>	Ordem inversa	<code>\$reverse([1, 2, 3])</code>	<code>[3, 2, 1]</code>
<code>\$sort(array)</code>	Matriz de classificação	<code>\$sort([3, 1, 2])</code>	<code>[1, 2, 3]</code>
<code>\$distinct(array)</code>	Remover duplicações	<code>\$distinct([1, 2, 2, 3])</code>	<code>[1, 2, 3]</code>
<code>\$map(array, func)</code>	Aplique a função a cada elemento	<code>\$map([1,2,3], function(\$v) {\$v*2})</code>	<code>[2, 4, 6]</code>

Booleano (1)

Função	Description	Exemplo	Resultado
<code>\$not(value)</code>	Lógico NÃO	<code>\$not(false)</code>	<code>true</code>

Aleatório (1)

Função	Description	Exemplo	Resultado
<code>\$random()</code>	Número aleatório entre 0 (inclusive) e 1 (exclusivo)	<code>\$random() > 0.5 ? 'A' : 'B'</code>	"A" ou "B"

Note

`$random()` gera um novo valor em cada avaliação. Se você precisar do mesmo valor aleatório em várias chaves de saída, vincule-o primeiro a uma variável: (`$r := $random(); ...`).

Date/time (4)

Função	Description	Exemplo	Resultado
<code>\$now()</code>	Carimbo de data/hora atual como uma string ISO 8601	<code>\$now()</code>	"2024-01-15T12:00:00.000Z"
<code>\$millis()</code>	Carimbo de data/hora atual em milissegundos desde a época	<code>\$millis()</code>	1705320000000
<code>\$toMillis(string)</code>	Converter string ISO 8601 em milissegundos	<code>\$toMillis('2024-01-15T12:00:00.000Z')</code>	1705320000000
<code>\$fromMillis(number)</code>	Converter milissegundos em string ISO 8601	<code>\$fromMillis(1705320000000)</code>	"2024-01-15T12:00:00.000Z"

Codificação (6)

Função	Description	Exemplo
<code>\$encodeURIComponent(string)</code>	Codificação de URL (preserva caracteres estruturais como /, , ?) &	<code>\$encodeURIComponent('https://example.com/path?q=hello world')</code>
<code>\$encodeURIComponentComponent(string)</code>	O URL codifica um único componente (codifica todos os caracteres especiais)	<code>\$encodeURIComponent('a&b=c')</code> → "a%26b%3Dc"
<code>\$decodeUrl(string)</code>	Decodifique uma string URL-encoded	<code>\$decodeUrl('hello%20world')</code> → "hello world"

Função	Description	Exemplo
<code>\$decodeURIComponent(string)</code>	Decodificar um componente URL-encoded	<code>\$decodeURIComponent('a%26b') → "a&b"</code>
<code>\$base64encode(string)</code>	Codificar para Base64	<code>\$base64encode('hello') → "aGVsbG8="</code>
<code>\$base64decode(string)</code>	Decodifique a partir de Base64	<code>\$base64decode('aGVsbG8=') → "hello"</code>

Tip

Use `$encodeURIComponent()` para valores de parâmetros de consulta individuais. Use `$encodeUrl()` somente quando precisar codificar um URL completo, preservando sua estrutura.

Padrões comuns

Valores alternativos

Forneça um padrão quando um valor pode não existir.

```
{%$exists(player_params.region) ? player_params.region : 'us-east-1%'}
```

Construção dinâmica de URL

Crie uma URL do servidor de decisão de anúncios a partir de várias entradas.

```
{%'https://ads.example.com/v1/vast?ip=' & $encodeURIComponent(session.client_ip) & '&ua=' & $encodeURIComponent(session.user_agent) & '&sid=' & session.id%'}
```

O código de status verifica a saída HTTP_REQUEST

Proteja os valores de saída contra falhas de HTTP.

```
{%response.statusCode != null and response.statusCode = 200 ? response.body.envelope : 'default-envelope%'}
```

Conversão numérica dos parâmetros do player

Os parâmetros do player chegam como strings. Converta-os antes das comparações aritméticas ou numéricas.

```
{%$number(player_params.max_duration) > 30 ? 'long' : 'short'%}
```

Important

Se `$number()` receber uma string não numérica, ela retornará `undefined`. Combine-o com `$exists()` quando o parâmetro pode estar ausente ou inválido: `($val := $number(player_params.max_duration); $exists($val) and $val > 30 ? 'long' : 'short')`.

Divisão aleatória de tráfego

Atribua espectadores a grupos de experimentos usando `$random()`.

```
{%$random() > 0.5 ? 'https://ads.example.com/v1/vast-a' : 'https://ads.example.com/v1/vast-b'%}
```

Classificação do tipo de dispositivo

Classifique os dispositivos com base na string do agente do usuário.

```
{%$contains(session.user_agent, 'CTV') ? 'ctv' : $contains(session.user_agent, 'Mobile') ? 'mobile' : 'desktop'%}
```

Exemplos de funções

Antes de analisar esses exemplos, você deve estar familiarizado com:

- [Hooks do ciclo de vida](#)— Campos de entrada e namespaces de saída disponíveis em cada gancho do ciclo de vida.
- [Tipos de funções e composição](#)— Como cada tipo de função funciona e seus campos de configuração.

- [Criando e gerenciando](#)— Como criar funções e anexá-las às configurações de reprodução.

Esta página fornece configurações de funções completas e funcionais para casos de uso comuns. Cada exemplo inclui o cenário, a configuração completa, o mapeamento da função e uma explicação do que acontece quando a função é executada.

Exemplo	Descrição do cenário
the section called “Exemplo 1: enriquecimento de dados”	Obtenha um envelope de LiveRamp identidade e no início da sessão e armazene-o nos parâmetros do player.
the section called “Exemplo 2: divisão A/B de tráfego”	Divida o tráfego de solicitações de anúncios uniformemente entre dois URLs do servidor de decisão de anúncios para A/B testes.

Exemplo 1: enriquecimento de dados

Cenário

Um serviço de streaming quer buscar dados de identidade do público de um serviço externo (nesse caso, LiveRamp) no início da sessão e armazená-los nos parâmetros do player para segmentação de anúncios posteriormente na sessão. Um envelope de identidade é um identificador criptografado fornecido por um serviço de resolução de identidade que permite que os sistemas de anúncios reconheçam um espectador em vários dispositivos e plataformas sem expor informações pessoais.

Configuração

Buscar LiveRamp envelope (HTTP_REQUEST):

```
{
  "FunctionId": "fetchLiveRamp",
  "FunctionType": "HTTP_REQUEST",
  "HttpRequestConfiguration": {
    "Runtime": "JSONATA",
    "MethodType": "GET",
    "Url": "{%'https://api.example-identity.com/v2/envelope?pid=12345&iv=' &
player_params.hashId%}"
```

```
    "Headers": {
      "Authorization": "{%'Bearer ramp-token-789'%}"
    },
    "RequestTimeoutMilliseconds": 2000,
    "Output": {
      "player_params.envelope_id": "{%response.body.envelopes[0].value%}"
    }
  }
}
```

Note

`Bearer ramp-token-789` Substitua por suas próprias credenciais de API. Não armazene tokens confidenciais diretamente nas configurações de função na produção. Considere usar uma estratégia de rotação de tokens.

Defina `RequestTimeoutMilliseconds` com base no tempo de resposta esperado do seu serviço externo. Um valor menor reduz a latência, mas aumenta a chance de erros de tempo limite.

Mapeamento de funções

```
{
  "FunctionMapping": {
    "PRE_SESSION_INITIALIZATION": "fetchLiveRamp"
  }
}
```

O que acontece quando a função é executada

1. Um espectador inicia uma sessão de reprodução.
2. MediaTailor executa o gancho do `PRE_SESSION_INITIALIZATION` ciclo de vida e funciona. `fetchLiveRamp`
3. A função cria o URL da solicitação usando `player_params.hashId` e chama a LiveRamp API. No URL, `pid` é o ID do parceiro e `iv` é o identificador do visualizador com hash.
4. A expressão de saída extrai o valor do primeiro envelope na matriz de resposta (`response.body.envelopes[0].value`) e o grava em `player_params.envelope_id`

 Tip

Para lidar com erros de forma explícita, verifique `response.statusCode` antes de acessar os dados de resposta: `{%response.statusCode = 200 ? response.body.envelopes[0].value : ''%}`

Para ter mais informações, consulte [HTTP_REQUEST](#), [INICIALIZAÇÃO DA PRÉ-SESSÃO](#), [Solução de problemas e monitoramento](#) e [Limites](#).

Exemplo 2: divisão A/B de tráfego

Cenário

Um serviço de streaming quer usar uma função `CUSTOM_OUTPUT` para dividir aleatoriamente o tráfego de solicitações de anúncios entre dois URLs do servidor de decisão de anúncios (ADS) para testes. A/B Metade das solicitações de anúncios vai para o endpoint v1 e metade vai para o endpoint v2.

Configuração

A/B divisão de tráfego (`CUSTOM_OUTPUT`):

```
{
  "FunctionId": "abTestAdsUrl",
  "FunctionType": "CUSTOM_OUTPUT",
  "CustomOutputConfiguration": {
    "Runtime": "JSONATA",
    "Output": {
      "adsRequest.url": "{%$random() < 0.5 ? 'https://ads.example.com/v1/decision?session=' & session.id : 'https://ads.example.com/v2/decision?session=' & session.id%}"
    }
  }
}
```

Nesta configuração:

- `FunctionId`— Um nome exclusivo para essa função.
- `FunctionType`— `CUSTOM_OUTPUT` avalia expressões sem fazer chamadas HTTP.

- **Output**— Mapeia `adsRequest.url` para uma expressão que seleciona aleatoriamente um dos dois URLs do ADS.

Mapeamento de funções

```
{
  "FunctionMapping": {
    "PRE_ADS_REQUEST": "abTestAdsUrl"
  }
}
```

O que acontece quando a função é executada

1. Um intervalo publicitário é encontrado durante a reprodução.
2. MediaTailor executa o gancho do `PRE_ADS_REQUEST` ciclo de vida e funciona. `abTestAdsUrl`
3. A `$random()` função retorna um valor entre 0 e 1. Se o valor for menor que 0,5, a função define o URL do ADS como o endpoint v1. Caso contrário, ele define o URL para o endpoint v2.

A `$random()` função é avaliada em cada intervalo de anúncio, portanto, a divisão é por solicitação de anúncio, não por sessão.

Tip

Para ajustar a taxa de divisão de tráfego, altere o valor limite. Por exemplo, `$random() < 0.8` envia 80% do tráfego para o primeiro URL e 20% para o segundo.

Note

Escrever em `adsRequest.url` substitui o URL padrão do ADS configurado na configuração de reprodução do intervalo publicitário atual.

Para obter mais informações, consulte [SAÍDA_PERSONALIZADA](#), [SOLICITAÇÃO PRÉVIA DE ANÚNCIOS](#) e [Referência de expressão JSonata](#).

Funções de solução de problemas e monitoramento

Esta página ajuda você a diagnosticar erros comuns de função e monitorar o desempenho da função na produção. A seção de solução de problemas é organizada por sintoma — comece com o que você observa, depois siga a causa e corrija.

Monitoramento

CloudWatch métricas

MediaTailor publica métricas para execução de funções na Amazon CloudWatch. Não é necessário optar por participar.

Hook-level métricas — um ponto de dados por execução do gancho do ciclo de vida:

Métrica	Description	Dimensões
PreSessionInitHook. .Invocations	Contagem de execuções por gancho	ConfigurationName
PreSessionInitHook. .Errors	Contagem de erros de gancho	ConfigurationName
PreSessionInitHook. .Latency	Tempo de execução do gancho (ms)	ConfigurationName
PreAdsRequestHook. .Invocations	Contagem de execuções por gancho	ConfigurationName
PreAdsRequestHook. .Errors	Contagem de erros de gancho	ConfigurationName
PreAdsRequestHook. .Latency	Tempo de execução do gancho (ms)	ConfigurationName

Function-level métricas — um ponto de dados por execução de função individual:

Métrica	Description	Dimensões
Function.Invocations	Contagem de execuções de funções	ConfigurationName , FunctionId , FunctionT ype , HookType
Function.Errors	Contagem de erros de função	ConfigurationName , FunctionId , FunctionT ype , HookType
Function.Latency	Tempo de execução da função (ms)	ConfigurationName , FunctionId , FunctionT ype , HookType

Para obter mais detalhes sobre como configurar alarmes e trabalhar com essas métricas, consulte [Monitoramento AWS Elemental MediaTailor com CloudWatch métricas da Amazon](#).

Eventos de log

MediaTailor emite eventos de log para execução da função. Os eventos de erro são emitidos por padrão. Eventos concluídos e resumidos são opcionais.

Tipo de evento	Padrão/ Opt-in	Grupo de registros	Description
PRE_SESSION_INIT_HOOK_SUMMARY	Opt-in	Registro do manifesto	Resumo da execução do gancho (success/error)
PRE_SESSION_INIT_HOOK_ERROR	Padrão	Registro do manifesto	Falha no gancho com errorType e causa
PRE_SESSION_INIT_FUNCTION_COMPLETED	Opt-in	Registro do manifesto	Função individual concluída com input/output

Tipo de evento	Padrão/ Opt-in	Grupo de registros	Description
PRE_SESSION_INIT_FUNCTION_ERROR	Padrão	Registro do manifesto	Falha de função individual
PRE_ADS_REQUEST_HOOK_SUMMARY	Opt-in	Registro de interação do ADS	Resumo da execução do gancho (success/error)
PRE_ADS_REQUEST_HOOK_ERROR	Padrão	Registro de interação do ADS	Falha no gancho com errorType e causa
PRE_ADS_REQUEST_FUNCTION_COMPLETED	Opt-in	Registro de interação do ADS	Função individual concluída com input/output
PRE_ADS_REQUEST_FUNCTION_ERROR	Padrão	Registro de interação do ADS	Falha de função individual

Para habilitar eventos de registro opcionais, consulte [Monitoramento AWS Elemental MediaTailor com CloudWatch métricas da Amazon](#).

Use o eventId campo para correlacionar eventos em nível de gancho e em nível de função para a mesma execução.

A seguinte consulta do Amazon CloudWatch Logs Insights filtra eventos de erro de função eventId para rastrear uma única execução:

```
fields @timestamp, eventType, functionId, errorType, cause
| filter eventId = "5dc6f040-0f72-4e8c-a64e-25eeef62708c"
| sort @timestamp asc
```

Solução de problemas

Quando uma função falha, MediaTailor registra um `errorType` campo no evento de erro. Use esse campo para identificar a classe de falha:

Tipo de erro	Description
SYNTAX_ERROR	A expressão falhou ao compilar ou encontrou um erro de tipo de tempo de execução
RESOURCE_LIMIT_ERROR	A expressão excedeu os limites de tempo de CPU, memória ou profundidade da pilha
RESTRICTION_ERROR	A expressão usou uma função bloqueada ou o limite de tamanho da carga de entrada foi excedido
TIMEOUT_ERROR	A execução da função excedeu seu limite de tempo
VALIDATION_ERROR	O caminho de saída visa um campo não gravável no escopo do gancho atual
INTERNAL_ERROR	Falha na infraestrutura não relacionada à função

As entradas são organizadas por sintoma e fazem referência a esses tipos de erro quando aplicável.

A expressão retorna null inesperadamente

Sintoma: Um valor de saída que você esperava que fosse preenchido está `null` ou está ausente nos parâmetros do player.

Causas possíveis:

Causa	Como identificar	Correção
O campo de entrada não existe nesse gancho do ciclo de vida.	Você referenciou <code>adsRequest.url</code> em uma <code>PRE_SESSION_INITIALIZATION</code> função. Os dados da solicitação do ADS não estão disponíveis no início da sessão.	Mova a função para o gancho <code>PRE_ADS_REQUEST</code> do ciclo de vida ou use um campo de entrada diferente. Consulte Hooks do ciclo de vida .
O campo de entrada está ausente dos dados da sessão.	Você fez referência a <code>player_params.campaign_id</code> , mas o player não passou esse parâmetro na inicialização da sessão.	Use <code>\$exists()</code> para verificar antes de acessar: <code>{% \$exists(player_params.campaign_id) ? player_params.campaign_id : 'default' %}</code> .
Você escreveu um objeto ou matriz nos parâmetros do player ou na solicitação do ADS.	Esses namespaces aceitam somente strings, números e booleanos. Objetos e matrizes são filtrados.	Armazene dados complexos em <code>temp.*</code> e extraia sequências de caracteres, números ou booleanos em uma etapa subsequente.

RESOURCE_LIMIT_ERROR: Stack overflow

Sintoma: A função falha com `errorType: "RESOURCE_LIMIT_ERROR"` cause: "Stack overflow error" e.

Causa: A expressão excedeu a profundidade máxima da pilha de 100 níveis. Isso normalmente acontece com expressões condicionais (`if/then/else`) profundamente aninhadas ou atribuições de variáveis complexas.

Isso significa que a expressão tem muitos níveis de aninhamento MediaTailor para ser processada.

Correção: simplifique a expressão. Divida a lógica complexa em várias entradas de saída ou várias etapas em um executor sequencial.

RESOURCE_LIMIT_ERROR: tempo limite da CPU

Sintoma: A função falha com `errorType: "RESOURCE_LIMIT_ERROR"` cause: "Expression evaluation timeout: Check for infinite loop" e.

Causa: A expressão excedeu o limite de tempo de CPU de 100 ms. Isso pode acontecer com expressões que realizam cálculos caros em grandes estruturas de dados.

Correção: reduza a complexidade da expressão. Se você estiver processando matrizes grandes, considere mover essa lógica para um serviço externo e chamá-la com uma `HTTP_REQUEST` função.

RESTRICTION_ERROR: Função não permitida

Sintoma: A função falha com `errorType: "RESTRICTION_ERROR"` cause: "Function '<name>' is not allowed" e.

Causa: A expressão chama uma função `JSonata` que não está na lista permitida de 38 funções. Exemplos comuns incluem `$filter$reduce`, `$eval$split`, `$join` e.

Correção: verifique o `cause` campo para ver o nome da função bloqueada. Substitua-o por uma alternativa permitida. Veja [Referência de expressão JSonata](#) a lista completa das 38 funções permitidas.

As funções permitidas comumente usadas incluem `$string` `$number` `$substring`, `$contains`, `$encodeURIComponent` e.

RESTRICTION_ERROR: Expressão muito longa

Sintoma: A função não consegue ser criada ou atualizada com `cause: "Expression length <actual> exceeds limit <limit>"`.

Causa: Uma única expressão excede 1.000 caracteres.

Correção: divida a expressão em partes menores. Use várias entradas de saída ou divida a lógica em várias etapas em um executor sequencial. Use vinculação variável (`:=`) para evitar a repetição de subexpressões longas.

Erro HTTP: o código de status é nulo

Sintoma: Na saída de uma `HTTP_REQUEST` função, `response.statusCode` é `null`.

Causa: a API externa estava inacessível, a conexão atingiu o tempo limite ou ocorreu um erro na rede. Quando isso acontece, MediaTailor define `response.statusCode` como `nullnull`, `response.body` para e `response.text` para "Internal Error".

Correção: sempre verifique `response.statusCode` antes de acessar os dados de resposta:

```
{%response.statusCode = 200 ? response.body.value : 'default%'}
```

Essa expressão verifica se a chamada HTTP retornou um código de status 200. Em caso afirmativo, ele usa o valor da resposta. Caso contrário, ele volta para um valor padrão.

Se isso acontecer com frequência, verifique se a API externa está íntegra. Considere aumentar `RequestTimeoutMilliseconds` se a API estiver lenta ou diminuí-la se você quiser falhar rapidamente.

Erro HTTP: o corpo da resposta é nulo

Sintoma: `response.statusCode` é 200, mas `response.body` é `null`.

Causa: o corpo da resposta não é um JSON válido ou tem mais de 20.000 caracteres. MediaTailor analisa apenas respostas JSON de até 20.000 caracteres. `response.body`

Correção: use `response.text` como alternativa. O `response.text` campo contém o corpo bruto da resposta truncado para 20.000 caracteres:

```
{%response.statusCode = 200 ? ($exists(response.body.id) ? response.body.id :  
$substring(response.text, 0, 100)) : 'error%'}
```

Se os dados de que você precisa estiverem além do limite de 20.000 caracteres, considere pedir à API externa que retorne uma resposta menor (por exemplo, solicitando campos específicos).

Erro HTTP: falha na validação do URL

Sintoma: A função falha em tempo de execução com uma mensagem sobre a URL estar malformada, usando um esquema inválido ou excedendo o tamanho máximo.

Causas possíveis:

Causa	Correção
O URL não usa <code>http</code> ou <code>https</code> .	Certifique-se de que a expressão do URL produza um URL começando com <code>http://</code> ou <code>https://</code> .
O URL excede 2.048 caracteres após a avaliação da expressão.	Encurte o URL. Mova valores de parâmetros grandes para o corpo da solicitação usando um método POST.
O URL está incorreto (não é um URI válido).	Verifique se há caracteres ausentes ou extras na expressão. Use <code>\$encodeURIComponent()</code> para consultar valores de parâmetros que podem conter caracteres especiais.

VALIDATION_ERROR

Sintoma: A função falha com `errorType: "VALIDATION_ERROR"`. Esse erro pode ocorrer no momento da criação (quando você cria ou atualiza a função) ou no momento da execução (quando a função é executada durante uma sessão).

Causas possíveis:

Causa	Exemplo	Correção
A chave de saída tem como alvo um namespace não gravável no gancho atual.	Escrevendo <code>adsRequest.url</code> em uma função anexada <code>PRE_SESSION_INITIALIZATION</code> a.	Verifique quais namespaces de saída são permitidos em seu gancho de ciclo de vida. <code>PRE_SESSION_INITIALIZATION</code> só permite <code>player_params.*</code> . Mova a função para o gancho correto ou altere a chave de saída. Consulte Hooks do ciclo de vida .

Causa	Exemplo	Correção
A chave de saída usa caracteres inválidos.	Uma chave de saída como <code>player_params.device type</code> (com um espaço). Somente letras, números, sublinhados e hífen são permitidos.	Renomeie a chave de saída para usar somente caracteres válidos. Por exemplo, use <code>player_params.device_type</code> em vez disso.
A chave de saída não começa com um prefixo válido.	Uma chave de saída como <code>custom.myValue</code> em vez de <code>player_params.myValue</code> ou <code>temp.myValue</code> .	Use um prefixo de saída válido: <code>player_params.*</code> , <code>temp.*</code> , ou <code>squadsRequest.*</code> .
Uma expressão JSonata tem um erro de sintaxe.	Uma citação de fechamento ausente ou uma condição incompleta: <code>{%session.id & %}</code> .	Examine a expressão em busca de aspas ausentes, parênteses sem correspondência ou operadores sem suporte, como <code>ou. ?? ? :</code>
Falta um campo obrigatório em uma função HTTP_REQUEST.	O campo URL está vazio ou o método não foi especificado.	Certifique-se de que os campos de URL e método estejam definidos. O método deve ser GET ou POST.
O URL criado pela expressão é inválido.	O URL avaliado usa um esquema não suportado <code>ftp://</code> , como, por exemplo, excede 2.048 caracteres ou está incorreto.	Verifique se a expressão do URL produz um <code>https://</code> URL <code>http://</code> ou válido. Use <code>\$encodeURIComponent()</code> para consultar valores de parâmetros que podem conter caracteres especiais.

Causa	Exemplo	Correção
Um cabeçalho HTTP contém caracteres inválidos ou usa um nome restrito.	Um valor de cabeçalho contém quebras de linha ou o nome do cabeçalho é <code>host-outransfer-encoding</code> .	Remova caracteres inválidos dos valores do cabeçalho. Evite nomes de cabeçalho s restritos. Consulte HTTP_REQUEST os limites do cabeçalho.

Verifique o `cause` campo no evento de registro de erros — ele identifica qual campo ou expressão falhou na validação.

INTERNAL_ERROR

Sintoma: A função falha com `errorType: "INTERNAL_ERROR"`.

Causa: Ocorreu uma falha na infraestrutura que não está relacionada à configuração da função.

Correção: repita a solicitação. Se o erro persistir, entre em contato com o AWS Support.

Limites de funções

Esta página lista os limites que se aplicam às Funções. Esses limites são aplicados no momento da criação (quando você cria ou atualiza uma função) e no tempo de execução (quando a função é executada).

Limites de execução do Hook

Gancho	Limite	Valor
PRE_SESSION_INITIALIZATION	Tempo limite total	2.000 ms
PRE_SESSION_INITIALIZATION	<code>player_params</code> Tamanho total	Mil caracteres
PRE_ADS_REQUEST	Tempo limite total	2.000 ms

Esses tempos limite abrangem toda a execução do gancho, incluindo todas as etapas da função e chamadas HTTP. Se o gancho exceder seu tempo limite, MediaTailor descarta toda a saída e procede como se nenhuma função estivesse conectada.

Os tempos limite de funções individuais (como `RequestTimeoutMilliseconds` para funções `HTTP_REQUEST`) devem caber no tempo limite total do gancho. Por exemplo, se o tempo limite do gancho for de 2.000 ms e uma função `HTTP_REQUEST` for `RequestTimeoutMilliseconds` definida como 2.000 ms, a função poderá expirar antes de ser concluída se algum processamento ocorrer antes ou depois da chamada HTTP.

O tamanho `player_params` total inclui a contagem combinada de caracteres de todos os nomes e valores das chaves de saída. Por exemplo, produzindo `player_params.deviceType = "mobile"` e `player_params.region = "us-east-1"` contandodeviceType, mobileregion, e chegando us-east-1 ao total de 1.000 caracteres.

Limites de expressão

Limite	Valor	Aplicado em
Tempo de CPU por expressão	100 ms	Runtime
Profundidade da pilha	100	Runtime
Comprimento da expressão	Mil caracteres	Autoria

Quando uma expressão excede um limite em tempo de execução, MediaTailor interrompe a expressão e registra o erro. Para obter detalhes sobre tipos de erro específicos, consulte [Solução de problemas e monitoramento](#).

Limites da composição da função

Limite	Valor
Etapas por executor sequencial	1 – 10
Profundidade máxima de nidificação	2
Total de execuções de funções por gancho	20

Limite	Valor
Entradas de saída por função	20
Referências circulares	Não permitido

Esses limites são aplicados no momento da criação.

Limites de solicitação HTTP

Limite	Valor
Tamanho do URL	2,048 caracteres
Solicitar tamanho corporal	64 KB
Contagem de cabeç	50
Comprimento do nome do cabeçalho	256 caracteres
Comprimento do valor do cabeçalho	8.192 caracteres
Tempo limite da solicitação	100—2.000 ms (via configuração pelo cliente) RequestTimeoutMilliseconds
Esquemas de URL permitidos	https, http
Cabeçalhos restritos	Host, Transfer-Encoding , Content-L ength , Connection

O comprimento do URL, o tamanho do corpo, a contagem de cabeçalhos e os limites de tamanho do cabeçalho são aplicados no momento da criação. O tempo limite da solicitação e os cabeçalhos restritos são aplicados em tempo de execução.

Se uma função definir um cabeçalho restrito, MediaTailor aceita a configuração da função no momento da criação, mas descarta o cabeçalho quando a solicitação HTTP for enviada durante a execução.

Limites de resposta

Limite	Valor
<code>response.body</code> tamanho máximo	20.000 caracteres
<code>response.text</code> comprimento máximo	20.000 caracteres

Esses limites são aplicados em tempo de execução.

Limites de saída

Limite	Valor
Comprimento máximo do valor de saída (por chave)	Mil caracteres

Esses limites são aplicados em tempo de execução.

Funções JSONata permitidas

Para ver a lista completa das funções permitidas, consulte [Referência de expressão JSONata](#).

Para obter ajuda no diagnóstico de erros relacionados a esses limites, consulte [Solução de problemas e monitoramento](#).

MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS

AWS Elemental MediaTailor usa variáveis dinâmicas de anúncios para passar informações da sua sessão de visualização para o servidor de decisão de anúncios (ADS). Essas informações ajudam o ADS a selecionar os anúncios mais relevantes para seus espectadores.

Esta seção fornece uma visão geral das variáveis dinâmicas do anúncio e links para guias de implementação específicos. Para obter instruções de step-by-step configuração, consulte os tópicos individuais abaixo.

Tipos de variáveis dinâmicas

MediaTailor suporta quatro tipos de variáveis dinâmicas:

- Variáveis de sessão — valores gerados automaticamente, como ID da sessão e dados SCTE-35. Consulte [MediaTailor variáveis de sessão para solicitações de ADS](#).
- Variáveis do player — Parâmetros personalizados enviados pelo seu player de vídeo. Consulte [MediaTailor variáveis de jogador para solicitações de ADS](#).
- Variáveis de domínio com aliases de configuração — Domínios de URL dinâmicos para configurações de várias origens.
- Aliases de configuração — mapeamentos predefinidos para substituição dinâmica de variáveis. Consulte [Aliases de configuração](#).

Casos de uso comuns

Use variáveis de anúncios dinâmicos para:

- Passe os dados demográficos e as preferências dos espectadores para seu ADS
- Encaminhe solicitações para origens diferentes com base na localização geográfica
- Permita a visualização com mudança de horário com integração MediaPackage
- Implemente A/B cenários de teste e failover

As seções a seguir fornecem detalhes adicionais sobre o uso de variáveis dinâmicas de anúncios com MediaTailor.

Tópicos

- [MediaTailor variáveis de sessão para solicitações de ADS](#)
- [MediaTailor variáveis de jogador para solicitações de ADS](#)
- [MediaTailor variáveis de domínio para várias fontes de conteúdo](#)
- [MediaTailor visão geral dos aliases de configuração](#)
- [MediaTailor passando parâmetros para o ADS](#)
- [MediaTailor roteamento de parâmetros para ADS e origens](#)
- [MediaTailor e integração de MediaPackage visualização com mudança de horário](#)
- [MediaTailor comportamento e persistência da sessão de parâmetros](#)

- [MediaTailor referência e limitações de parâmetros](#)
- [MediaTailor guia de solução de problemas de parâmetros](#)
- [MediaTailor guia de solução de problemas de aliases de configuração](#)

Para requisitos de formatação de parâmetros e solução de problemas, consulte [MediaTailor referência e limitações de parâmetros](#) e [MediaTailor guia de solução de problemas de parâmetros](#)

MediaTailor variáveis de sessão para solicitações de ADS

AWS Elemental MediaTailor envia dados da sessão para o Ad Decision Server (ADS) quando você configura AWS Elemental MediaTailor para especificar uma ou mais das variáveis listadas nesta seção no modelo de URL do ADS. Você pode usar variáveis individuais e concatenar várias variáveis para criar um único valor. MediaTailor gera alguns valores e obtém o restante de fontes como o manifesto e a solicitação de inicialização da sessão do player.

A tabela a seguir descreve as variáveis de dados da sessão que você pode usar em seu modelo de configuração de URL de solicitação do ADS. Os números das seções listados na tabela correspondem à versão 2019a da especificação -35 da Society of Cable Telecommunications Engineers (SCTE) -35, [Digital Program Insertion Cueing Message](#). Para obter detalhes sobre a [pré-busca de anúncios](#), consulte [Pré-busca de anúncios](#)

Nome	Disponível para pré-busca de anúncios	Seção de especificação SCTE-35	Description
[avail.index]	Sim		Um número que representa a posição de um anúncio disponível em um índice. No início de uma sessão de reprodução, MediaTailor cria um índice de todos os anúncios disponibilizados em um manifesto e armazena o índice para o restante da sessão. Quando MediaTailor faz uma solicitação ao ADS para preencher a disponibilidade, ela inclui o número do índice de disponibilidade do anúncio. Esse parâmetro permite que o ADS melhore a seleção de anúncios usando recursos como exclusão competitiva e limitação de frequência.

Nome	Disponível para pré-busca de anúncios	Seção de especificação SCTE-35	Description
[avail.random]	Sim		Um número aleatório entre 0 e 10.000.000.000, como um número longo, que é MediaTailor gerado para cada solicitação ao ADS. Alguns servidores de anúncios usam esse parâmetro para habilitar recursos como separar anúncios de empresas concorrentes.
[scte.archive_allowed_flag]	Sim	10.3.3.1	Um valor booleano opcional. Quando esse valor é 0, as restrições de gravação são declaradas no segmento. Quando esse valor é 1, as restrições de gravação não são declaradas no segmento.
[scte.avail_num]	Sim	9.7.2.1	O valor analisado pelo MediaTailor campo SCTE-35avail_num , como um número longo. MediaTailor Eu posso usar esse valor para designar números lineares e disponíveis. O valor deve ser um número inteiro.
[scte.avails_expected]	Sim	9,7.2.1	Um valor longo opcional que fornece a contagem esperada de disponibilidades no evento atual.
[scte.delivery_not_restricted_flag]	Sim	10.3.3.1	Um valor booleano opcional. Quando esse valor é 0, os próximos cinco bits são reservados. Quando esse valor é 1, os próximos cinco bits assumem os significados descritos na especificação SCTE-35.
[scte.device_restrictions]	Sim	10.3.3.1	Um valor inteiro opcional que sinaliza três grupos de dispositivos predefinidos, independentes e não hierárquicos. Para obter mais informações sobre essa variável, consulte a descrição segments_expected na especificação SCTE-35.

Nome	Disponível para pré-busca de anúncios	Seção de especificação SCTE-35	Description
[scte.event_id]	Sim	9.1 e 9.7.2.1	O valor analisado pelo MediaTailor campo SCTE-35splice_event_id , como um número longo. MediaTailor usa esse valor para designar números lineares de disponibilidade de anúncios ou para preencher cadeias de caracteres de consulta do servidor de anúncios, como posições do pod de anúncios. O valor deve ser um número inteiro.
[scte.no_regional_blackout_flag]	Sim	10.3.3.1	Um valor booleano opcional. Quando esse valor é 0, as restrições regionais de blackout se aplicam ao segmento. Quando esse valor é 1, as restrições regionais de blackout não se aplicam ao segmento.
[scte.segment_num]	Sim	10.3.3.1	Um valor inteiro opcional que numera segmentos dentro de uma coleção de segmentos. Para obter mais informações sobre essa variável, consulte a descrição segment_num na especificação SCTE-35.
[scte.segmentation_event_id]	Sim	10.3.3.1	MediaTailor expõe essa variável como scte.event_id .
[scte.segmentation_type_id]	Sim	10.3.3.1	Um valor inteiro opcional de 8 bits que especifica o tipo de segmentação. Para obter mais informações sobre essa variável, consulte a descrição segmentation_type_id na especificação SCTE-35.

Nome	Disponível para pré-busca de anúncios	Seção de especificação SCTE-35	Description
[scte.segmentation_upid]	segmentation_upid_type : Sim private_data : Sim	segmentation_upid: 10.3.3.1 UPID privado gerenciado: 10.3.3.3	Corresponde ao elemento SCTE-35segmentation_upid . O segmentation_upid elemento contém segmentation_upid_type , segmentation_upid_length e. MediaTailor suporta os seguintes segmentation_upid tipos: • Informações sobre ADS (0x0E) - Informações publicitárias. Para obter mais informações, consulte a descrição de segmentation_upid na especificação SCTE-35. • UPID privado gerenciado (0x0C) - A estrutura do UPID privado gerenciado (MPU) conforme definido na especificação SCTE-35. MediaTailor suporta representações SCTE binárias ou DASH XML. Você pode usar essa estrutura em um fluxo de trabalho de podbuster. Para fazer isso, especifique 32 bits (4 bytes) format_identifier e inclua os seguintes parâmetros no private_data atributo: <pre>ABCD{"assetId":" my_program ", "cueData": {"cueType":" theAdType ", "key":" pb", "value": " 123456"}}</pre> MediaTailor analisa os valores do JSON anterior e os passa para as variáveis scte.segmentation_upid.assetId dinâmicas scte.segmentation_upid.cueData.key , e scte.segmentation_upid.cueData.value

Nome	Disponível para pré-busca de anúncios	Seção de especificação SCTE-35	Description
			Definido pelo usuário (0x01) - Uma estrutura definida pelo usuário. Para obter mais informações, consulte a descrição de <code>segmentation_upid</code> na especificação SCTE-35.
[scte.segmentation_upid.assetId]	Sim		Usado em conjunto com o Managed Private UPID (0xC) <code>segmentation_upid_type</code> para fluxos de trabalho de podbuster. MediaTailor deriva esse valor do <code>assetId</code> parâmetro na estrutura JSON do MPU. <code>private_data</code> Para obter mais informações, consulte Managed Private UPID JSON structure for a podbuster workflow .
[scte.segmentation_upid.cueData.key]	Sim		Usado em conjunto com o Managed Private UPID (0xC) <code>segmentation_upid_type</code> para fluxos de trabalho de podbuster. MediaTailor deriva esse valor do <code>cueData.key</code> parâmetro na estrutura JSON do MPU. <code>private_data</code> Para obter mais informações, consulte Managed Private UPID JSON structure for a podbuster workflow .
[scte.segmentation_upid.cueData.value]	Sim		Usado em conjunto com o Managed Private UPID (0xC) <code>segmentation_upid_type</code> para fluxos de trabalho de podbuster. MediaTailor deriva esse valor do <code>cueData.key</code> parâmetro na estrutura JSON do MPU. <code>private_data</code> Para obter mais informações, consulte Managed Private UPID JSON structure for a podbuster workflow. O valor pode ser uma string.

Nome	Disponível para pré-busca de anúncios	Seção de especificação SCTE-35	Description
[scte.segmentation_upid.private_data.{index}]	Sim		Usado em conjunto com o UPID privado gerenciado (0xC) <code>segmentation_upid_type</code> para fluxos de trabalho de publicidade direcionada. MediaTailor divide tokens UPID de segmentação delimitados por dois pontos e cria variáveis de sessão indexadas. O índice corresponde à posição na lista delimitada por dois pontos, ignorando o espaço em branco inicial dos dois pontos iniciais. Por exemplo, se <code>segmentation_upid = ":3213214:2313321/5:3943"</code>, então: <code>[scte.segmentation_upid.private_data.0] = 3213214</code> <code>[scte.segmentation_upid.private_data.1] = 2313321/5</code> <code>[scte.segmentation_upid.private_data.2] = 3943</code> O valor pode ser uma string.
[scte.segments_expected]	Sim	10.3.3.1	Um valor inteiro opcional que fornece a contagem esperada de segmentos individuais em uma coleção de segmentos. Para obter mais informações sobre essa variável, consulte a descrição <code>segments_expected</code> na especificação SCTE-35.
[scte.sub_segment_num]	Sim	10.3.3.1	Um valor inteiro opcional que identifica um subsegmento específico dentro de uma coleção de subsegmentos. Para obter mais informações sobre essa variável, consulte a descrição do <code>sub_segment_num</code> na especificação SCTE-35.

Nome	Disponível para pré-busca de anúncios	Seção de especificação SCTE-35	Description
[scte.sub_segments_expected]	Sim	10.3.3.1	Um valor inteiro opcional que fornece a contagem esperada de subsegmentos individuais em uma coleção de subsegmentos. Para obter mais informações sobre essa variável, consulte a descrição <code>sub_segments_expected</code> na especificação SCTE-35.
[scte.unique_program_id]	Sim	9.7.2.1	O valor inteiro analisado pelo campo <code>MediaTailor splice_insert SCTE-35.unique_program_id</code>. O ADS usa o ID exclusivo de programa (UPID) para fornecer direcionamento de anúncios em nível de programa para streamings lineares ao vivo. Se o comando SCTE-35 não for inserção de emenda, MediaTailor define-o como um valor vazio. O valor deve ser um número inteiro.

Nome	Disponível para pré-busca de anúncios	Seção de especificação SCTE-35	Description
[session.avail_duration_ms]	Sim		A duração em milissegundos do espaço de disponibilidade do anúncio. O valor padrão é 300.000 ms. AWS Elemental MediaTailor obtém o valor da duração do manifesto de entrada da seguinte forma: • Para HLS: MediaTailor obtém a duração dos <code>#EXT-X-CUE-OUT: DURATION</code> ou dos valores na <code>#EXT-X-DATERANGE</code> tag. Se o manifesto de entrada tiver uma duração nula, inválida ou 0 para o anúncio disponível nessas tags, MediaTailor use o padrão. • Para DASH: MediaTailor obtém o valor da duração da duração do evento, se for especificado. Do contrário, ele usa o valor padrão. • Para VOD: quando um stream de VOD aciona uma chamada de anúncio pré-listada, se o manifesto não incluir mensagens SCTE com um valor de duração, MediaTailor não insira uma duração para a <code>[session.avail_duration_ms]</code>, incluindo o valor de duração padrão.
[session.avail_duration_seconds]	Sim		A duração em segundos do espaço de disponibilidade do anúncio, ou disponibilidade do anúncio, arredonda para o segundo mais próximo. MediaTailor determina esse valor da mesma forma que determina <code>[session.avail_duration_ms]</code>.
[session.client_ip]	Não		O endereço IP remoto de onde veio a MediaTailor solicitação. Caso o cabeçalho <code>X-forwarded-for</code> esteja definido, esse valor é o usado pelo MediaTailor no <code>client_ip</code>.

Nome	Disponível para pré-busca de anúncios	Seção de especificação SCTE-35	Description
[session.id]	Não		Um identificador numérico exclusivo para a sessão de reprodução atual. Como todas as solicitações feitas por um player para uma sessão têm o mesmo ID, ele pode ser usado em campos ADS que devem correlacionar solicitações de uma única exibição.
[session.referer]	Não		Normalmente, o URL da página que está hospedando o player de vídeo. MediaTailor define essa variável com o valor do Referer cabeçalho que o player usou em sua solicitação para MediaTailor. Caso o player não forneça esse cabeçalho, o MediaTailor deixa o [session.referer] vazio. Se você usa uma rede de distribuição de conteúdo (CDN) ou proxy na frente do endpoint do manifesto e quiser que essa variável apareça, faça o proxy do cabeçalho correto do player aqui.
[session.user_agent]	Não		O User-Agent cabeçalho MediaTailor recebido da solicitação de inicialização da sessão do jogador. Caso esteja usando uma CDN ou um proxy à frente do endpoint do manifesto, você deve adicionar o cabeçalho correto do player aqui.
[session.uuid]	Não		Alternativa [session.id] a. Este é um identificador exclusivo para a sessão de reprodução atual, como o seguinte: e039fd39-09f0-46b2-aca9-9871cc116cde

Nome	Disponível para pré-busca de anúncios	Seção de especificação SCTE-35	Description
[avail.source_content_time_epoch_ms]	Não		Para HLS, o valor é o PDT do segmento de origem que iniciou a disponibilidade. Para DASH, o valor é o <code><SupplementalProperty> urn:scte:dash:utc-time</code> do <code><Period></code> que contém o <code><EventStream></code> - Para anúncios pré-lançados, HLS ou DASH <code>[avail.source_content_time_epoch_ms]</code> é o PDT do primeiro segmento HLS e o do primeiro <code><SupplementalProperty> urn:scte:dash:utc-time <Period></code> Para transmissões ao vivo com janelas de manifesto curtas, esse valor será diferente para cada espectador com base em quando eles começarem a assistir à transmissão. Para uma transmissão ao vivo com uma janela completa de DVR de eventos, o valor será o mesmo para cada espectador. - Para disponibilizações parciais devido à supressão de disponibilidade, <code>[avail.source_content_time_epoch_ms]</code> é o PDT do segmento de conteúdo de origem que iniciou a disponibilização. Por exemplo, se MediaTailor remover os primeiros 20 segundos da disponibilidade, ainda <code>[avail.source_content_time_epoch_ms]</code> será configurado para o PDT original, não alterado em 20 segundos. - Para solicitações de pré-busca do ADS, <code>[avail.source_content_time_epoch_ms]</code> está vazio, no entanto, os clientes podem configurá-lo usando variáveis dinâmicas de recuperação do cronograma de pré-busca.

Example

Caso o ADS exija um parâmetro de consulta chamado `deviceSession` a ser passado com o identificador da sessão exclusivo, o URL ADS do modelo em AWS Elemental MediaTailor pode se parecer com o seguinte:

```
https://my.ads.server.com/path?deviceSession=[session.id]
```

AWS Elemental MediaTailor gera automaticamente um identificador exclusivo para cada fluxo e insere o identificador no lugar de `session.id`. Se o identificador for `1234567`, a solicitação final MediaTailor feita ao ADS seria mais ou menos assim:

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

Se o ADS exigir que vários parâmetros de consulta sejam transmitidos, o modelo de URL do ADS AWS Elemental MediaTailor poderá ter a seguinte aparência:

```
https://my.ads.server.com/sample?
e=[scte.avails_expected]&f=[scte.segment_num]&g=[scte.segments_expected]&h=[scte.sub_segment_num]
```

O seguinte exemplo de fragmento XML do marcador DASH mostra como usar: `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>
</Period>
```

O seguinte exemplo de fragmento XML do marcador DASH mostra como usar: `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>

```

O seguinte exemplo de fragmento XML do marcador DASH mostra como usar: `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>QW5vdGhlciB0ZXN0IHN0cmVuZyBmb3IgZW5jb2RpbmcgdG8gQmFzZTY0IGVuY29kZWQgYmluYXJ5Lg==
Binary>
      </scte35:Signal>
    </Event>

```

O exemplo de tag HLS a seguir mostra como usar `EXT-X-DATERANGE`:

```

#EXT-X-DATERANGE:ID="splice-6FFFFFF0",START-DATE="2014-03-05T11:
15:00Z",PLANNED-DURATION=59.993,SCTE35-OUT=0xFC002F0000000000FF0
00014056FFFFFF000E011622DCAFF000052636200000000000A0008029896F50
0000087000000000

```

O exemplo de tag HLS a seguir mostra como usar `EXT-X-CUE-OUT`:

```
#EXT-OATCLS-SCTE35:/DA0AAAAAAAAAAAAABQb+ADAQ6QAeAhxDVUVJQAAA03/PAAEUrEoICAAAAAAg
+2UBNAAANvrtoQ==
#EXT-X-ASSET:CAID=0x00000000020FB6501
#EXT-X-CUE-OUT:201.467
```

O exemplo de tag HLS a seguir mostra como usar EXT-X-SPLICEPOINT-SCTE35:

```
#EXT-X-SPLICEPOINT-SCTE35:/DA9AAAAAAAAP/wBQb+uYbZqwAnAiVDVUVJAAAKqX//
AAEjW4AMEU1EU05CMDAXMTMyMjE5M190NAAAmXz5JA==
```

O exemplo a seguir mostra como usar a scte35:Binary decodificação:

```
{
  "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
}
}

```

MediaTailor variáveis de jogador para solicitações de ADS

AWS Elemental MediaTailor envia dados recebidos do player para o ADS quando você configura AWS Elemental MediaTailor para especificar `player_params.<query_parameter_name>` variáveis no URL do ADS do modelo. Por exemplo, se o player enviar um parâmetro de consulta nomeado `user_id` em sua solicitação para MediaTailor, para transmitir esses dados na solicitação do ADS, inclua-o [`player_params.user_id`] na configuração do URL do ADS.

Isso permite controlar os parâmetros de consulta incluídos na solicitação ADS. Normalmente, você adiciona um parâmetro de consulta especial reconhecido pelo ADS ao URL da solicitação ADS e fornece pares de chave/valor, como o valor do parâmetro.

Os exemplos usados no seguinte procedimento usam os seguintes pares de chave/valor:

- param1 com um valor value1:
- param2 com um valor value2:

Para adicionar parâmetros de consulta como pares de chave/valor

1. Em AWS Elemental MediaTailor, configure a URL do modelo de solicitação do ADS para referenciar os parâmetros. O seguinte URL mostra a inclusão dos parâmetros de exemplo:

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

2. (Opcional) Para relatórios de rastreamento de anúncios no lado do servidor, codifique o URL dos pares de chave/valor no player. Quando MediaTailor recebe a solicitação de inicialização da sessão, ele decodifica os valores por URL uma vez antes de substituí-los na URL da solicitação do ADS.

 Note

Caso o ADS exija um valor codificado por URL, o URL codifica o valor duas vezes no player. Dessa forma, a decodificação feita por MediaTailor resulta em um valor codificado uma vez para o ADS.

Por exemplo, caso a representação decodificada dos valores enviados ao ADS seja `param1=value1:&param2=value2:`, a representação codificada por URL é `param1=value1%3A&param2=value2%3A`.

3. Na chamada de inicialização da sessão do player, passe os pares de valores-chave para MediaTailor como o valor de um único parâmetro de consulta. As chamadas de exemplo a seguir fornecem os pares de chave/valor de exemplo para relatórios de rastreamento de anúncios nos lados do servidor e do cliente.
- Solicitações de exemplo para relatórios de rastreamento de anúncios no lado do servidor – usar pares codificados em URL

HLS:

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

DASH:

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

- Solicitação de exemplo para relatórios de rastreamento de anúncios no lado do cliente – sem codificação por URL

HLS:

```
POST <master>.m3u8
{
  "adsParams": {
    "param1": "value1:",
    "param2": "value2:"
  }
}
```

DASH:

```
POST <manifest>.mpd
{
  "adsParams": {
    "param1": "value1:",
    "param2": "value2:"
  }
}
```

Para relatórios do lado do servidor, MediaTailor decodifica os parâmetros quando a solicitação do jogador é recebida. Para relatórios do lado do cliente, isso não altera os parâmetros recebidos na carga JSON. MediaTailor envia a seguinte solicitação para o ADS:

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

Dessa maneira, o param1 e os pares de chave/valor param2 são incluídos como parâmetros de consulta de primeira classe na solicitação ADS.

MediaTailor variáveis de domínio para várias fontes de conteúdo

AWS Elemental MediaTailor variáveis dinâmicas de domínio permitem que você use vários domínios, como a parte my-ads-server.com da URL http://my-ads-server.com, com os parâmetros do player em sua configuração. Isso possibilita que você use mais de uma fonte de conteúdo ou servidor de decisão de anúncios (ADS) em uma única configuração.

Você pode usar variáveis de domínio com qualquer parâmetro que contenha um URI:

- AdDecisionServerUrl

- `AdSegmentUrlPrefix`
- `ContentSegmentUrlPrefix`
- `LivePreroll.AdDecisionServerUrl`
- `VideoContentSourceUrl`

As variáveis de domínio são usadas junto com os aliases de configuração para realizar a substituição dinâmica de variáveis. Os aliases de configuração mapeiam um conjunto de aliases e valores para os parâmetros do player que são usados para a configuração dinâmica do domínio. Para obter os procedimentos de configuração, consulte [Criação e uso de aliases de configuração com MediaTailor](#). Para obter informações de referência detalhadas, consulte [MediaTailor visão geral dos aliases de configuração](#).

MediaTailor visão geral dos aliases de configuração

AWS Elemental MediaTailor os aliases de configuração permitem a substituição dinâmica de variáveis em domínios de URL e outros campos compatíveis. Use esse recurso para usar vários domínios e configurar dinamicamente URLs durante a inicialização da sessão.

Casos de uso

Os aliases de configuração permitem arquiteturas sofisticadas de várias configurações para os seguintes cenários:

- Roteamento geográfico: direcione solicitações para diferentes origens ou servidores de anúncios com base na localização do espectador usando aliases específicos da região. Para obter orientação de implementação, consulte [CloudFront Origin Failover](#).
- Roteamento baseado em conteúdo: direcione diferentes tipos de conteúdo para origens especializadas ou pipelines de processamento. Para a configuração do comportamento de roteamento, consulte [the section called “Configurar comportamentos de roteamento de CDN”](#).
- Cenários de failover: implemente origens de backup e servidores de anúncios com failover automático usando alternância de aliases. Para uma implementação detalhada, consulte [Implemente resiliência multirregional com o MQAR MediaTailor the section called “Planeje a integração com a CDN”](#) e.
- Teste A/B: teste diferentes servidores de anúncios, origens ou configurações roteando o tráfego com base nos parâmetros do player. Para obter orientação sobre testes de carga, consulte [Preparar e executar testes de desempenho para a Amazon CloudFront com monitoramento real de usuários](#).

- Otimização específica do dispositivo: otimize a entrega de conteúdo e a veiculação de anúncios para diferentes tipos ou recursos de dispositivos. Para obter uma orientação abrangente, consulte [Configurar a filtragem de manifestos com MediaTailor MediaPackage, e CDN](#).
- Balanceamento de carga: distribua a carga em várias origens ou servidores de anúncios usando roteamento dinâmico. Para obter orientação de implementação, consulte [Implemente resiliência multirregional com o MQAR MediaTailor the section called “Planeje a integração com a CDN”](#) e.

Campos compatíveis

Você pode usar variáveis dinâmicas nos seguintes campos de configuração:

- VideoContentSourceUrl
- AdDecisionServerUrl
- LivePreroll.AdDecisionServerUrl
- AdSegmentUrlPrefix
- ContentSegmentUrlPrefix
- TranscodeProfileName
- SlateAdUrl
- StartUrl
- EndUrl

As seções a seguir descrevem como usar aliases de configuração.

Tópicos

- [Criação e uso de aliases de configuração com MediaTailor](#)
- [Alias de configuração com exemplo de MediaTailor uso](#)

Criação e uso de aliases de configuração com MediaTailor

Antes de começar a usar variáveis de domínio, você cria aliases de configuração para sua configuração. Você usa os aliases de configuração como variáveis de substituição de domínio no momento da inicialização da sessão.

Restrições

Observe as seguintes restrições ao usar aliases de configuração:

- Todas as variáveis dinâmicas usadas no domínio devem ser definidas como variáveis `ConfigurationAliases` dinâmicas.
- As variáveis dos parâmetros do player devem ser prefixadas `complayer_params..` Por exemplo, `.player_params.origin_domain`
- A lista de valores com alias deve ser exaustiva para variáveis de domínio em estado crítico URLs (`VideoContentSourceUrl`, `AdSegmentUrlPrefix`). `ContentSegmentUrlPrefix`
- Se for feita uma solicitação para uma variável de domínio crítica URLs que não especifique a variável dinâmica ou use um alias inválido, a solicitação falhará com um código de 400 status HTTP. Campos não críticos (`SlateAdUrl`, `TranscodeProfileName`, bumper URLs) registrarão os avisos, mas não falharão na solicitação.

Comportamento alternativo para aliases ausentes

Quando os aliases de configuração não são encontrados ou são inválidos, MediaTailor implementa o seguinte comportamento de fallback:

- Variáveis de domínio: se um alias de variável de domínio estiver ausente ou for inválido, a solicitação falhará com o código de status HTTP 400. Todas as variáveis de domínio devem ter aliases válidos definidos.
- Variáveis que não são de domínio: para variáveis usadas em partes que não são de domínio URLs (como elementos de caminho ou parâmetros de consulta), aliases ausentes resultam na substituição de uma string vazia.
- Validação de configuração: MediaTailor valida se todos os aliases necessários estão presentes durante as operações de criação e atualização da configuração.

Etapa 1: criar aliases de configuração

Para criar aliases de configuração a serem usados para substituição de domínio usando o MediaTailor console, execute o procedimento a seguir.

Console

Para criar aliases de configuração usando o console

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.

2. Na seção Apelidos de configuração na página Configurações, escolha Adicionar parâmetro de player.
3. Em Parâmetro do jogador, insira o nome do parâmetro do jogador que você deseja usar como uma variável dinâmica. Por exemplo, `.player_params.origin_domain`
4. Em Aliases, insira os aliases e seus valores que você deseja usar para o parâmetro do player.
5. Escolha OK.

AWS Elemental MediaTailor exibe o novo parâmetro na tabela na seção Apelidos de configuração.

6. Repita as etapas anteriores para adicionar mais parâmetros do player.
7. Escolha Salvar.

API

Para criar aliases de configuração usando a API

Ao criar ou atualizar uma MediaTailor configuração, use o `ConfigurationAliases` parâmetro com a seguinte estrutura JSON:

```
{
  "ConfigurationAliases": {
    "player_params.origin_domain": {
      "pdx": "abc.mediapackage.us-west-2.amazonaws.com",
      "iad": "xyz.mediapackage.us-east-1.amazonaws.com"
    },
    "player_params.ad_type": {
      "customized": "abc12345",
      "default": "defaultAdType"
    }
  }
}
```

Etapa 2: Usar aliases de configuração na inicialização da sessão

Depois de configurar os aliases de configuração, você pode usá-los como variáveis de substituição para domínios na solicitação de inicialização da sessão. Isso permite que você configure dinamicamente os domínios da sua sessão.

Example Exemplo de aliases de configuração básica

Aqui está um exemplo básico de uma configuração que inclui aliases de configuração e variáveis dinâmicas de domínio:

```
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"
    }
  }
}
```

Example Inicialização da sessão com aliases

Usando a configuração anterior, uma solicitação de inicialização de sessão usando as variáveis e os aliases do player seria semelhante à seguinte:

```
POST index.m3u8
{
  "playerParams": {
    "origin_domain": "pdx",
    "region": "pdx",
    "endpoint_id": "pdx",
```

```
    "ad_type": "customized"
  }
}
```

MediaTailor substitui as cadeias de caracteres de aliases pelos valores mapeados na configuração de aliases de configuração.

A solicitação para o ADS terá a seguinte aparência:

```
https://abc.execute-api.us-west-2.amazonaws.com/ads?sid=[session.id]&ad_type=abc12345
```

A solicitação à origem dos manifestos terá a seguinte aparência:

```
https://abc.mediapackage.us-west-2.amazonaws.com/out/v1/abcd
```

Alias de configuração com exemplo de MediaTailor uso

Os exemplos a seguir mostram como é feita uma MediaTailor configuração completa com aliases de configuração, uma solicitação de inicialização de sessão com aliases e o fluxo de processamento de aliases.

Example Configuração completa com aliases

O exemplo a seguir mostra uma configuração completa que inclui aliases de configuração e variáveis dinâmicas de domínio:

```
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]",

  "AdSegmentUrlPrefix": "https://[player_params.ad_cdn_domain]/ads/",
  "ContentSegmentUrlPrefix": "https://[player_params.content_cdn_domain]/content/",
  "TranscodeProfileName": "[player_params.transcode_profile]",
  "SlateAdUrl": "https://[player_params.slate_domain]/slate/[player_params.slate_type].mp4",
  "StartUrl": "https://[player_params.tracking_domain]/start?session=[session.id]",
  "EndUrl": "https://[player_params.tracking_domain]/end?session=[session.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"
  },
  "player_params.ad_cdn_domain": {
    "pdx": "ads-west.cdn.example.com",
    "iad": "ads-east.cdn.example.com"
  },
  "player_params.content_cdn_domain": {
    "pdx": "content-west.cdn.example.com",
    "iad": "content-east.cdn.example.com"
  },
  "player_params.transcode_profile": {
    "mobile": "mobile_optimized",
    "desktop": "high_quality",
    "tv": "4k_profile"
  },
  "player_params.slate_domain": {
    "pdx": "slate-west.example.com",
    "iad": "slate-east.example.com"
  },
  "player_params.slate_type": {
    "standard": "default_slate",
    "branded": "brand_slate"
  },
  "player_params.tracking_domain": {
    "pdx": "tracking-west.example.com",
    "iad": "tracking-east.example.com"
  }
}
```

```
}
```

Example Inicialização da sessão com aliases

O exemplo a seguir mostra uma solicitação de inicialização de sessão que especifica as variáveis e os aliases do player:

```
POST master.m3u8
{
  "playerParams": {
    "origin_domain": "pdx",
    "region": "pdx",
    "endpoint_id": "pdx",
    "ad_type": "customized",
    "ad_cdn_domain": "pdx",
    "content_cdn_domain": "pdx",
    "transcode_profile": "mobile",
    "slate_domain": "pdx",
    "slate_type": "branded",
    "tracking_domain": "pdx"
  }
}
```

Example Fluxo de processamento de parâmetros

No exemplo a seguir, MediaTailor substitui as cadeias de caracteres de alias pelos valores mapeados nos aliases de configuração. O processamento resulta nas seguintes solicitações:

- Solicitação de ADS:

```
https://abc.execute-api.us-west-2.amazonaws.com/ads?sid=[session.id]&ad_type=abc12345
```

- VideoContentSource solicitação:

```
https://abc.mediapackage.us-west-2.amazonaws.com/out/v1/abcd
```

- AdSegmentUrlPrefix:

```
https://ads-west.cdn.example.com/ads/
```

- ContentSegmentUrlPrefix:

```
https://content-west.cdn.example.com/content/
```

- TranscodeProfileName:

```
mobile_optimized
```

- SlateAdUrl:

```
https://slate-west.example.com/slate/brand_slate.mp4
```

- StartUrl:

```
https://tracking-west.example.com/start?session=[session.id]
```

- EndUrl:

```
https://tracking-west.example.com/end?session=[session.id]
```

MediaTailor passando parâmetros para o ADS

AWS Elemental MediaTailor suporta a configuração de variáveis dinâmicas em MediaTailor solicitações ao ADS usando as etapas a seguir.

- Para obter informações sobre a formatação compatível com os parâmetros de consulta, consulte [MediaTailor referência e limitações de parâmetros](#).
- Para aliases de configuração e variáveis de domínio, consulte [MediaTailor visão geral dos aliases de configuração](#).
- Para obter personalizações adicionais na solicitação do ADS, consulte. [Uso avançado](#)

Métodos de inicialização da sessão

MediaTailor suporta vários métodos para inicialização de sessão e passagem de parâmetros:

1. POST com corpo de solicitação:

```
POST <master>.m3u8
{
  "adsParams": {"param1": "value1", "param2": "value2"},
```

```
"playerParams": {"param3": "value3"}
}
```

2. Parâmetros de consulta no URL:

```
GET <master>.m3u8?ads.param1=value1&ads.param2=value2&playerParams.param3=value3
```

Important

Você só pode especificar os parâmetros uma vez, no momento da inicialização. Os aliases de configuração são resolvidos para valores reais antes do encaminhamento.

Para passar informações de sessão e player ao ADS

1. Trabalhe com o ADS para determinar as informações de que ele precisa para responder a uma consulta de anúncio AWS Elemental MediaTailor.
2. Crie uma configuração MediaTailor que use um modelo de URL de solicitação do ADS que atenda aos requisitos do ADS. No URL, inclua parâmetros estáticos e espaços reservados para parâmetros dinâmicos. Digite o URL do modelo no campo Ad decision server (Servidor de decisões de anúncios) da configuração.

No seguinte URL do modelo de exemplo, `correlation` fornece dados da sessão e `deviceType` fornece dados do player:

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

3. No player, configure a solicitação de iniciação da sessão para o AWS Elemental MediaTailor fornecer parâmetros para os dados do player. Inclua os parâmetros na solicitação de iniciação da sessão e os omita de solicitações subsequentes da sessão.

O tipo de chamada que o jogador faz para inicializar a sessão determina se o jogador (cliente) ou MediaTailor (servidor) fornece relatórios de rastreamento de anúncios para a sessão. Para obter informações sobre essas duas opções, consulte [Dados de relatórios e rastreamento](#).

Faça um dos tipos de chamadas a seguir, dependendo do desejo de relatórios para o rastreamento de anúncios no lado do servidor ou do cliente. Em ambas as chamadas de exemplo, `userID` é desejado para o ADS e `auth_token` é desejado para a origem:

- (Opção) Solicite relatórios de rastreamento de anúncios do lado do servidor — Prefixe os parâmetros com os quais você deseja enviar MediaTailor ao ADS. `ads` Deixe o prefixo desativado para parâmetros para os quais o MediaTailor deve enviar ao servidor de origem:

Os exemplos a seguir mostram as solicitações recebidas de HLS e DASH para AWS Elemental MediaTailor. MediaTailor usa o `deviceType` em sua solicitação para o ADS e o `auth_token` em sua solicitação para o servidor de origem.

Exemplo de HLS:

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

Exemplo de DASH:

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

- (Opção) Solicite relatórios de rastreamento de anúncios do lado do cliente — Forneça parâmetros para o ADS dentro de um objeto. `adsParams`

Exemplo de HLS:

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

Exemplo de DASH:

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



```
}
```

Quando o jogador inicia uma sessão, AWS Elemental MediaTailor substitui as variáveis no URL de solicitação do ADS modelo pelos dados da sessão e pelos parâmetros do ads jogador. Ele passa os parâmetros restantes do player para o servidor de origem.

Example MediaTailor solicitações com variáveis de anúncio

Os exemplos a seguir mostram as chamadas para o ADS e o servidor de origem do AWS Elemental MediaTailor que correspondem aos exemplos de chamada de inicialização de sessão do player anterior:

- MediaTailor chama o ADS com os dados da sessão e o tipo de dispositivo do jogador:

```
https://my.ads.server.com/path?correlation=896976764&deviceType=ipad
```

- MediaTailor chama o servidor de origem com o token de autorização do jogador.

- Exemplo de HLS:

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

- Exemplo de DASH:

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

Uso avançado

Personalize a solicitação ADS de muitas maneiras com dados do player e da sessão. Você só precisa incluir o nome do host do ADS.

Os exemplos a seguir mostram algumas das maneiras de personalizar a solicitação:

- Concatene os parâmetros do player e da sessão para criar novos parâmetros. Exemplo:

```
https://my.ads.com?key1=[player_params.value1][session.id]
```

- Use um parâmetro de player como parte de um elemento do caminho. Exemplo:

```
https://my.ads.com/[player_params.path]?key=value
```

- Use os parâmetros do player para passar ambos os elementos de caminho e as chaves propriamente ditas, e não apenas valores. Exemplo:

```
https://my.ads.com/[player_params.path]?[player_params.key1]=[player_params.value1]
```

MediaTailor roteamento de parâmetros para ADS e origens

AWS Elemental MediaTailor encaminha os parâmetros de consulta para destinos diferentes com base em seu prefixo e finalidade. Compreender o roteamento de parâmetros é essencial para implementar funcionalidades específicas de origem, como visualização com mudança de horário com. MediaPackage

Regras de roteamento de parâmetros

MediaTailor usa as seguintes regras de roteamento para parâmetros de consulta:

- Parâmetros de origem (sem prefixo): parâmetros sem um prefixo específico são passados para o servidor de origem para funcionalidade específica da origem
- Parâmetros do ADS (**ads.**prefixo): os parâmetros prefixados com `ads.` são enviados para o Ad Decision Server
- Parâmetros do manifesto (**manifest.**prefixo): os parâmetros prefixados com `manifest.` são usados para roteamento e autorização de CDN

Exemplo de roteamento de parâmetros

A inicialização da sessão a seguir demonstra o roteamento de parâmetros:

```
POST /v1/session/123456789/originId/index.m3u8
{
  "adsParams": {
    "param1": "value1",
    "param2": "value2"
  },
  "manifestParams": {
    "auth_token": "abc123"
  }
}
```

Neste exemplo:

- `param1` e `param2` são enviados para o ADS
- `auth_token` é usado para roteamento e autorização de CDN
- Parâmetros sem prefixos seriam passados para o servidor de origem

Comportamento dos parâmetros do servidor de origem

Os parâmetros passados aos servidores de origem permitem funcionalidades específicas da origem, como visualização com mudança de horário, filtragem de conteúdo e autenticação.

Casos de uso de parâmetros de origem comuns

Os parâmetros de origem são comumente usados para:

- Visualização com mudança de horário: `start` e `end` parâmetros para MediaPackage conteúdo com mudança de horário
- Autenticação de conteúdo: tokens de autenticação exigidos pelo servidor de origem
- Filtragem de conteúdo: parâmetros que controlam quais variantes de conteúdo são retornadas
- Recursos específicos da origem: quaisquer parâmetros que o servidor de origem usa para processamento de conteúdo

Important

Os parâmetros são processados na inicialização da sessão e mantidos durante toda a sessão. Para modificar parâmetros como janelas de mudança de horário, você deve criar uma nova sessão com valores atualizados.

MediaTailor e integração de MediaPackage visualização com mudança de horário

AWS Elemental MediaTailor pode passar parâmetros de visualização com mudança de horário para MediaPackage as origens para permitir a funcionalidade de visualização inicial e atualização. Essa integração permite que os espectadores comecem a assistir ao conteúdo ao vivo de momentos anteriores.

MediaPackage parâmetros de visualização com mudança de tempo

MediaPackage suporta os seguintes parâmetros de visualização com mudança de horário que podem ser transmitidos: MediaTailor

- **start**: Carimbo de data e hora ISO 8601 que define o início do manifesto com mudança de horário
- **end**: data e hora da época ou ISO 8601 definindo o final do manifesto com mudança de horário
- **time_delay**: atrasar a disponibilidade do conteúdo em segundos especificados
- **manifest_window_seconds**: solicite um manifesto menor que a janela configurada

Example MediaTailor inicialização da sessão com parâmetros com mudança de MediaPackage horário

O exemplo a seguir mostra como inicializar uma sessão com parâmetros de visualização com mudança de horário:

```
GET /v1/master/123456789/originId/index.m3u8?  
start=2024-08-26T10:00:00Z&end=2024-08-26T11:00:00Z
```

Ou usando a inicialização explícita da sessão:

```
POST /v1/session/123456789/originId/index.m3u8  
{  
  "adsParams": {  
    "param1": "value1"  
  }  
}
```

Com parâmetros de consulta adicionais:

```
?start=2024-08-26T10:00:00Z&end=2024-08-26T11:00:00Z
```

Comportamento dos parâmetros durante as sessões

Os parâmetros de visualização com mudança de horário têm características de comportamento específicas:

- **Inicialização da sessão**: os parâmetros são processados quando a sessão é criada
- **Persistência de parâmetros**: os parâmetros permanecem associados à sessão durante a reprodução

- Imutável após a inicialização: os parâmetros não podem ser alterados durante uma sessão ativa
- Nova sessão necessária: para modificar as janelas de mudança de horário, crie uma nova sessão com valores de parâmetros atualizados

MediaPackage requisitos da janela de partida

Para que a visualização por mudança de horário funcione MediaPackage, certifique-se do seguinte:

1. Configure uma janela de inicialização em seu MediaPackage endpoint (até 24 horas)
2. Certifique-se de que sua CDN encaminhe os parâmetros de consulta necessários para MediaPackage
3. Use janelas de reprodução consistentes em todas as sessões do player para melhorar o cache da CDN
4. Verifique se os horários de início e término estão dentro da janela de inicialização configurada

Important

Ao usar a visualização com mudança de horário, use janelas de reprodução consistentes em todas as sessões do player, em vez de gerar horários de início ou término exclusivos para cada espectador. Isso gera um melhor armazenamento em cache na CDN e evita possíveis limitações.

Para obter informações completas sobre a configuração e os parâmetros MediaPackage da visualização com mudança de horário, consulte [Visualização com mudança de horário AWS Elemental MediaPackage](#) no Guia do usuário.AWS Elemental MediaPackage

MediaTailor comportamento e persistência da sessão de parâmetros

AWS Elemental MediaTailor processa os parâmetros na inicialização da sessão e os mantém durante todo o ciclo de vida da sessão. Compreender o comportamento da sessão é crucial para implementar cenários de parâmetros dinâmicos.

Métodos de inicialização da sessão

MediaTailor suporta vários métodos para inicialização de sessão com parâmetros:

1. Inicialização implícita da sessão: parâmetros incluídos na solicitação inicial do manifesto

```
GET /v1/master/123456789/originId/index.m3u8?  
manifest.auth_token=abc123&start=2024-08-26T10:00:00Z
```

2. Inicialização explícita da sessão (POST): parâmetros fornecidos no corpo da solicitação

```
POST /v1/session/123456789/originId/index.m3u8  
{  
  "adsParams": {"param1": "value1"},  
  "manifestParams": {"auth_token": "abc123"}  
}
```

3. Inicialização explícita da sessão (GET): parâmetros fornecidos como parâmetros de consulta

```
GET /v1/session/123456789/originId/index.m3u8?  
ads.param1=value1&manifestParams.auth_token=abc123
```

Persistência e imutabilidade dos parâmetros

MediaTailor o comportamento dos parâmetros segue estas regras:

- Especificação única: os parâmetros só podem ser especificados uma vez, na inicialização da sessão
- Persistência em toda a sessão: os parâmetros são mantidos durante toda a sessão
- Imutável após a inicialização: os parâmetros não podem ser modificados após a criação da sessão
- Resolução do alias de configuração: os aliases são resolvidos para valores reais antes de serem encaminhados para os destinos

Cenários de modificação de parâmetros

Para modificar os parâmetros durante a reprodução:

- Criar nova sessão: inicie uma nova sessão com valores de parâmetros atualizados
- Transição do jogador: faça a transição perfeita do jogador para a nova sessão
- Herança de parâmetros: transfira parâmetros inalterados para manter a consistência

Example Modificando parâmetros de mudança de horário

Para mudar de uma janela de 1 hora para uma janela de 2 horas:

1. Sessão atual: `start=2024-08-26T10:00:00Z&end=2024-08-26T11:00:00Z`
2. Crie uma nova sessão: `start=2024-08-26T10:00:00Z&end=2024-08-26T12:00:00Z`
3. Transição do player para o novo URL da sessão

Important

Várias solicitações de playlist multivariantes para uma única sessão não atualizam os parâmetros após a primeira solicitação. Os parâmetros permanecem imutáveis durante a duração da sessão.

MediaTailor referência e limitações de parâmetros

Antes de configurar variáveis dinâmicas de anúncios, entenda os requisitos e limitações de formatação de parâmetros que se aplicam a todas as MediaTailor configurações.

AWS Elemental MediaTailor fornece informações abrangentes sobre restrições de caracteres de parâmetros, limitações de comprimento e formatos compatíveis tanto para parâmetros de consulta de manifesto quanto para parâmetros ADS.

Restrições de caracteres do parâmetro de consulta manifesto

Os parâmetros de consulta do manifesto oferecem suporte a caracteres específicos e têm limitações de comprimento definidas.

Caracteres compatíveis (sem codificação de URL)

Você pode usar os seguintes caracteres diretamente nos parâmetros de consulta do manifesto:

- Caracteres alfanuméricos (A-Z, a-z, 0-9)
- Períodos (.)
- Hífens (-)
- Sublinha (_)
- Barras invertidas (\)

Caracteres compatíveis com codificação de URL

Os seguintes caracteres especiais são suportados quando codificados por URL:

- período (.) = %2E
- traço (-) = %2D
- sublinhado (_) = %5F
- porcentagem (%) = %25
- tilde (~) = %7E
- barra para frente (/) = %2F
- asterisco (*) = %2A
- é igual a (=) = %3D
- pergunta (?) = %3F

Suporte para codificação de URL

MediaTailor suporta o sinal de porcentagem (%) quando você o usa na codificação de URL (por exemplo, `hello%20world` = hello world). Você pode usar qualquer caractere codificado em URL, desde que sejam codificações de URL válidas de acordo com a especificação HTTP.

Caracteres não suportados

Você não pode usar os seguintes caracteres nos parâmetros de consulta do manifesto sem a codificação de URL: `:`, `,`, `?`, `&`, `=`, `%`, `/` (barra invertida).

Important

MediaTailor não suporta caracteres duplos, como `%%%` ou `==`. Você não pode usar full URLs como valores de parâmetros de consulta de manifesto devido a restrições de caracteres.

Limitações de comprimento

O tamanho total de todos os parâmetros de consulta do manifesto (chaves e valores combinados) não deve exceder 2.000 caracteres.

Limitações de comprimento do parâmetro ADS

As seguintes limitações de comprimento se aplicam aos parâmetros usados nas solicitações ao ADS:

- Nome do parâmetro ADS: máximo de 10.000 caracteres
- Valor do parâmetro ADS: máximo de 25.000 caracteres
- URL do ADS: máximo de 25.000 caracteres

MediaTailor guia de solução de problemas de parâmetros

AWS Elemental MediaTailor fornece orientação para solucionar problemas comuns relacionados a parâmetros MediaTailor, incluindo restrições de caracteres, problemas de codificação de URL e erros de alias de configuração.

Erros de restrição de caracteres

Valores de parâmetros que contêm caracteres não suportados podem causar erros ou comportamento inesperado.

Sintomas comuns

Os sintomas a seguir podem indicar problemas de restrição de caracteres:

- Parâmetros que não aparecem no manifesto URLs
- Erros HTTP 400 durante a inicialização da sessão
- Valores de parâmetros truncados ou corrompidos
- Falha nas solicitações do ADS devido à malformação URLs

Etapas de resolução

Para resolver erros de restrição de caracteres:

1. Revise os valores dos parâmetros para caracteres não suportados: :, ?, &, =, % /
2. Aplique a codificação de URL adequada para caracteres especiais (consulte) [MediaTailor referência e limitações de parâmetros](#)
3. Evite caracteres duplos, como %%% ou ==
4. Considere formatos alternativos de parâmetros se não for URLs possível usar full

Exemplo Exemplo de codificação de URL

Em vez de usar:

```
manifest.redirect_url=https://example.com/path?param=value
```

Use o formato codificado por URL:

```
manifest.redirect_url=https%3A%2F%2Fexample.com%2Fpath%3Fparam%3Dvalue
```

Erros de limitação de comprimento

Os parâmetros que excedem os limites de comprimento podem ser truncados ou causar erros.

Limites de comprimento

Os seguintes limites de comprimento se aplicam (consulte [MediaTailor referência e limitações de parâmetros](#) para obter detalhes completos):

- Parâmetros de consulta do manifesto (total): 2000 caracteres
- Nomes de parâmetros do ADS: 10.000 caracteres
- Valores dos parâmetros do ADS: 25.000 caracteres
- ADS URLs: 25.000 caracteres

Estratégias de resolução

Para lidar com limitações de comprimento:

1. Use nomes e valores de parâmetros mais curtos sempre que possível
2. Divida valores de parâmetros grandes em vários parâmetros menores
3. Use aliases de configuração para mapear aliases curtos para valores mais longos (consulte [MediaTailor visão geral dos aliases de configuração](#))
4. Considere o uso de armazenamento externo para dados grandes com referências de parâmetros

Erros de alias de configuração

Problemas de alias de configuração podem resultar em erros HTTP 400 ou valores de parâmetros inesperados.

Erros comuns de alias de configuração

Os seguintes erros geralmente ocorrem com aliases de configuração:

- Erro HTTP 400: valor de alias ausente ou inválido
- Variáveis de domínio não estão sendo resolvidas corretamente
- Os parâmetros do jogador não estão sendo substituídos por valores de alias

Lista de verificação de resolução

Para resolver erros de alias de configuração:

1. Verifique se todas as variáveis de domínio estão definidas como `ConfigurationAliases`
2. Certifique-se de que as variáveis dos parâmetros do jogador usem o `player_params.` prefixo
3. Confirme se a lista de valores com alias é exaustiva para variáveis de domínio em estado crítico URLs (`VideoContentSourceUrl`,,) `AdSegmentUrlPrefix` `ContentSegmentUrlPrefix`
4. Verifique se as solicitações de inicialização da sessão especificam valores de alias válidos
5. Validar a estrutura JSON do parâmetro `ConfigurationAliases`

Para obter orientações detalhadas sobre solução de problemas, consulte [MediaTailor guia de solução de problemas de aliases de configuração](#).

Exemplo Validação do alias de configuração

Certifique-se de que sua configuração inclua todos os aliases necessários:

```
"ConfigurationAliases": {
  "player_params.origin_domain": {
    "pdx": "abc.mediapackage.us-west-2.amazonaws.com",
    "iad": "xyz.mediapackage.us-east-1.amazonaws.com"
    // Must include all possible values used in session initialization
  }
}
```

Problemas de fluxo de processamento de parâmetros

Compreender o fluxo de processamento de parâmetros ajuda a solucionar problemas com encaminhamento e transformação de parâmetros.

Ordem de processamento de parâmetros

MediaTailor processa os parâmetros na seguinte ordem:

1. Validação do parâmetro de inicialização da sessão
2. Resolução do alias de configuração (se aplicável)
3. Filtragem de parâmetros (ADS x origem x manifesto)
4. Codificação e formatação de URL
5. Aplicação de parâmetros para URLs

Fluxo de parâmetros de depuração

Para depurar problemas de processamento de parâmetros:

1. Verifique se os parâmetros foram especificados corretamente na inicialização da sessão
2. Verifique se os aliases de configuração são resolvidos para os valores esperados
3. Confirme se os parâmetros aparecem corretamente URLs (manifesto, ADS, origem)
4. Validar se a codificação do URL foi aplicada corretamente

Example Exemplo de fluxo de parâmetros

Inicialização da sessão:

```
POST master.m3u8
{
  "playerParams": {"origin_domain": "pdx"},
  "manifestParams": {"test": "123"}
}
```

Após a resolução e o processamento do alias:

- Solicitação de origem: `https://abc.mediapackage.us-west-2.amazonaws.com/out/v1/abcd`
- URL do manifesto: `/v1/master/.../index.m3u8?aws.sessionId=session&test=123`

Considerações de segurança e práticas recomendadas

MediaTailor implementa medidas de segurança para tratamento de parâmetros para evitar problemas comuns de segurança.

Medidas de segurança

MediaTailor implementa as seguintes medidas de segurança:

1. Limitações de tamanho de entrada para evitar inchaço no banco de dados
2. Codificação e higienização adequadas da entrada do usuário
3. Codificação de URL da entrada para evitar a corrupção da resposta

Práticas recomendadas

Siga estas práticas recomendadas para tratamento seguro de parâmetros:

- Valide os valores dos parâmetros no lado do cliente antes de enviar
- Use aliases de configuração para limitar possíveis valores de parâmetros
- Evite incluir informações confidenciais nos parâmetros
- Monitore o uso de parâmetros para padrões incomuns
- Mantenha os valores dos parâmetros dentro dos limites de comprimento recomendados

MediaTailor guia de solução de problemas de aliases de configuração

AWS Elemental MediaTailor fornece orientação sistemática de solução de problemas de aliases de configuração e cenários de erro.

Erros de validação do alias de configuração

Quando os aliases de configuração estão ausentes ou são inválidos, MediaTailor retorna respostas de erro específicas para ajudar a identificar o problema.

Cenários de erro comuns

A tabela a seguir descreve erros comuns de alias de configuração e suas etapas de resolução:

Erro	Causa	Resolução
HTTP 400: alias de parâmetro de jogador inválido	Valor do parâmetro do jogador não encontrado em ConfigurationAliases	Verifique se o valor do parâmetro do player existe como uma chave no

Erro	Causa	Resolução
		ConfigurationAliases mapeamento correspondente
HTTP 400: Falta o alias de configuração necessário	Variável de domínio usada sem ConfigurationAliases entrada correspondente	Adicione o parâmetro do player ausente a ConfigurationAliases com todos os mapeamentos de aliases necessários
HTTP 400: Falha na validação da configuração	ConfigurationAliases a estrutura está malformada ou incompleta	Valide a estrutura JSON e garanta que todas as variáveis de domínio tenham aliases correspondentes
Substituição de string vazia em URLs	Alias de variável não pertencente ao domínio não encontrado	Adicione o mapeamento de aliases ausente ou forneça o valor padrão em ConfigurationAliases

Lista de verificação de validação

Use a lista de verificação a seguir para validar sua configuração de aliases:

1. Cobertura de variáveis de domínio: garanta que todas as variáveis usadas em partes do domínio URLs tenham ConfigurationAliases entradas correspondentes
2. Completude do alias: verifique se todos os valores possíveis dos parâmetros do jogador estão incluídos como chaves nos mapeamentos de aliases
3. Estrutura JSON: valide se o ConfigurationAliases JSON está devidamente formatado e aninhado
4. Nomeação dos parâmetros: confirme se todos os parâmetros do player usam o prefixo `player_params`.
5. Consistência de valores: garanta que os valores de alias sejam válidos para o uso pretendido (URLs, nomes de perfil etc.)

Resolução de alias de configuração de depuração

Siga essa abordagem sistemática para depurar problemas de resolução de aliases de configuração.

Step-by-step metodologia de depuração

Use as etapas a seguir para identificar e resolver problemas de alias de configuração:

Procedimento de depuração de alias de configuração

1. Verifique a estrutura de configuração: confirme se sua configuração de reprodução inclui a formatação adequada ConfigurationAliases

```
{
  "ConfigurationAliases": {
    "player_params.example_param": {
      "alias1": "value1",
      "alias2": "value2"
    }
  }
}
```

2. Verifique o formato dos parâmetros do player: certifique-se de que a inicialização da sessão inclua parâmetros do player formatados corretamente

```
{
  "playerParams": {
    "example_param": "alias1"
  }
}
```

3. Validar o mapeamento de alias: confirme se o valor do parâmetro do jogador ("alias1") existe como uma chave no mapeamento ConfigurationAliases
4. Teste com configuração simples: comece com uma configuração mínima para isolar o problema
5. Monitore respostas de erro: verifique as respostas MediaTailor de erro para mensagens de validação específicas
6. Verificar se os resolvidos foram confirmados URLs: confirme se URLs os resolvidos finais são válidos e acessíveis

Práticas recomendadas de aliases de configuração

Siga essas práticas recomendadas para garantir a implementação confiável do alias de configuração.

Considerações sobre segurança

Implemente as seguintes medidas de segurança ao usar aliases de configuração:

- Validação de entrada: valide todos os valores dos parâmetros do jogador antes de usá-los na resolução de aliases
- Limpeza do valor do alias: certifique-se de que os valores do alias contenham somente caracteres e formatos esperados
- Restrições de domínio: limite os aliases de domínio a domínios confiáveis e controlados
- Controle de acesso: restrinja a modificação da configuração somente ao pessoal autorizado

Otimização de desempenho

Otimize o desempenho do alias de configuração com estas recomendações:

- Minimize a contagem de aliases: use somente os aliases necessários para reduzir a sobrecarga de processamento
- Nomenclatura eficiente: use convenções de nomenclatura claras e consistentes para aliases e parâmetros
- Valores padrão: forneça aliases padrão sensatos para casos de uso comuns
- Cache de configuração: aproveite MediaTailor o cache de configuração para melhorar o desempenho

Manutenção e monitoramento

Mantenha operações confiáveis de alias de configuração com estas práticas:

- Validação regular: valide periodicamente se todos os mapeamentos de aliases são atuais e funcionais
- Monitoramento de erros: monitore erros de HTTP 400 relacionados a aliases ausentes ou inválidos
- Documentação: mantenha uma documentação clara de todos os mapeamentos de aliases e suas finalidades
- Procedimentos de teste: implemente testes abrangentes para todas as combinações de aliases

MediaTailor parâmetros de consulta do manifesto

AWS Elemental MediaTailor manipula parâmetros de consulta para diferentes propósitos: parâmetros de consulta manifestos para roteamento e autorização de CDN e outros parâmetros de consulta que podem ser usados para funcionalidades específicas da origem.

AWS Elemental MediaTailor preserva os parâmetros de consulta da inicialização da sessão e os anexa ao manifesto personalizado URLs e a outros ativos. Use essa funcionalidade quando você tiver uma Rede de Distribuição de Conteúdo (CDN) entre MediaTailor e o player cliente.

Use parâmetros de consulta de manifesto quando sua CDN precisar dos parâmetros de consulta para o seguinte:

- Roteamento dinâmico para diferentes endpoints MediaTailor
- Autorização de token

Comportamento do lado do cliente versus CDN

MediaTailor acrescenta parâmetros de consulta para endpoints de relatórios do lado do cliente, mas não os acrescenta para segmentos de CDN. A funcionalidade atualizada fornece suporte mais abrangente para parâmetros de consulta em vários MediaTailor ativos, aumentando a flexibilidade para casos de uso de roteamento e autorização de CDN.

MediaTailor acrescenta parâmetros de consulta para endpoints de relatórios do lado do cliente, mas não anexa os parâmetros de consulta para os segmentos CloudFront (ou outros CDN).

Para usar a preservação de parâmetros, entre em contato com o [AWS Support](#) para solicitar que a passagem do parâmetro de consulta de manifesto seja ativada.

O comportamento varia entre HLS e DASH e entre a inicialização explícita e implícita da sessão. Os tópicos a seguir descrevem como configurar solicitações de inicialização de sessão para que MediaTailor passem pelos parâmetros para o manifesto:

MediaTailor tratamento de parâmetros de consulta para origens

AWS Elemental MediaTailor trata os parâmetros de consulta de forma diferente, dependendo de sua finalidade. Os parâmetros de consulta do manifesto (prefixados `commanifest.`) são usados para roteamento e autorização de CDN, enquanto outros parâmetros de consulta podem ser usados para funcionalidades específicas da origem.

Visualização com mudança de horário com MediaPackage

Para a funcionalidade de visualização com mudança de horário MediaPackage, você pode incluir `start` e `end` parâmetros em suas solicitações. Esses parâmetros definem janelas de conteúdo específicas para visualização inicial e atualização.

Exemplo solicitação de visualização com mudança de horário

Inclua parâmetros inicial e final em sua solicitação de manifesto para visualização com mudança de horário:

```
GET /v1/master/111122223333/originId/index.m3u8?  
start=2024-08-26T10:00:00Z&end=2024-08-26T11:00:00Z
```

Comportamento dos parâmetros durante as sessões

Os parâmetros de consulta são processados na inicialização da sessão. Para visualização com mudança de horário ou outros recursos específicos da origem:

- Inicialização da sessão: inclua os parâmetros necessários na solicitação inicial do manifesto
- Persistência de parâmetros: os parâmetros são associados à sessão e mantidos durante a reprodução
- Alteração de parâmetros: Para modificar janelas de mudança de horário ou outros parâmetros, crie uma nova sessão com valores atualizados

Important

O tratamento específico dos parâmetros de consulta depende da configuração de origem e dos parâmetros compatíveis com a origem do conteúdo. Para MediaPackage integração, certifique-se de que sua CDN esteja configurada para encaminhar os parâmetros de consulta necessários, conforme descrito em [Configurar parâmetros de consulta essenciais](#).

MediaTailor restrições de caracteres de parâmetros e codificação de URL

AWS Elemental MediaTailor suporta caracteres específicos nos parâmetros de consulta do manifesto. Você pode usar a codificação de URL para caracteres especiais.

Caracteres compatíveis com codificação de URL

Os seguintes caracteres especiais são compatíveis com a codificação de URL:

- período (.) = %2E
- traço (-) = %2D
- sublinhado (_) = %5F
- porcentagem (%) = %25
- tilde (~) = %7E
- barra para frente (/) = %2F
- asterisco (*) = %2A
- é igual a (=) = %3D
- pergunta (?) = %3F

Suporte para codificação de URL

MediaTailor suporta o sinal de porcentagem (%) quando você o usa na codificação de URL (por exemplo, hello%20world = hello world). Você pode usar qualquer caractere codificado em URL, desde que sejam codificações de URL válidas de acordo com a especificação HTTP.

Important

MediaTailor não suporta caracteres duplos, como%%% ou ==.

Considerações sobre segurança

MediaTailor implementa as seguintes medidas de segurança para tratamento de parâmetros:

1. Limitações de tamanho de entrada para evitar inchaço no banco de dados
2. Codificação e higienização adequadas da entrada do usuário
3. Codificação de URL da entrada para evitar a corrupção da resposta

Tópicos

- [MediaTailor Inicialização de sessão implícita do HLS](#)

- [MediaTailor Inicialização de sessão implícita do DASH](#)
- [MediaTailor Inicialização de sessão explícita HLS e DASH](#)
- [MediaTailor comportamento de parâmetros específicos do protocolo](#)
- [MediaTailor Integração CDN e roteamento de parâmetros](#)

MediaTailor Inicialização de sessão implícita do HLS

AWS Elemental MediaTailor inclui parâmetros de consulta em links para MediaTailor recursos quando sua solicitação inclui parâmetros de consulta com a chave `manifest.*`. O exemplo a seguir mostra esse formato de solicitação:

```
GET /v1/master/111122223333/originId/index.m3u8?manifest.test=123&other=456
```

Os links não incluem o `manifest.` prefixo.

Aplicação de parâmetros no HLS

Para sessões implícitas do HLS, MediaTailor aplica parâmetros aos seguintes locais na hierarquia do manifesto:

- Playlist multivariante URLs
- Playlist de mídia URLs
- Segmento de conteúdo URLs
- Segmento de anúncio URLs
- Inicialização do HLS URLs

Example playlist multivariante

O exemplo a seguir mostra como MediaTailor inclui os parâmetros de consulta na URL da playlist multivariante:

```
#EXTM3U
#EXT-X-VERSION:3
#EXT-X-INDEPENDENT-SEGMENTS
#EXT-X-MEDIA:LANGUAGE="eng",AUTOSELECT=YES,FORCED=NO,TYPE=SUBTITLES,URI=".../.../
manifest/111122223333/originId/session/1.m3u8?test=123",GROUP-
ID="subtitles",DEFAULT=YES,NAME="caption_1"
```

```
#EXT-X-STREAM-INF:CODECS="avc1.640029,mp4a.40.2",AVERAGE-
BANDWIDTH=2525657,RESOLUTION=960x540,SUBTITLES="subtitles",FRAME-
RATE=29.97,BANDWIDTH=2665212
../../manifest/111122223333/originId/session/0.m3u8?test=123
```

Example playlist de mídia

O exemplo a seguir mostra como MediaTailor incluir os parâmetros de consulta nos URLs segmentos de conteúdo:

```
#EXTM3U
#EXT-X-VERSION:6
#EXT-X-TARGETDURATION:7
#EXT-X-MEDIA-SEQUENCE:28716269
#EXT-X-DISCONTINUITY-SEQUENCE:0
#EXTINF:6.006,
https://origin.com/contentSegment_1.ts?originQueryParam=foo&test=123
#EXT-X-DISCONTINUITY
#EXTINF:6.006,
../../segment/111122223333/originId/session/0/2?test=123
```

MediaTailor Inicialização de sessão implícita do DASH

AWS Elemental MediaTailor cria uma sessão para o cliente e a redireciona com parâmetros de consulta quando o cliente faz uma solicitação de manifesto sem uma sessão. O exemplo a seguir mostra esse formato de solicitação:

```
GET /v1/dash/111122223333/originId/index.mpd?manifest.test=123&other=456
```

MediaTailor cria uma sessão para o cliente e a redireciona com os parâmetros de consulta:

```
/v1/dash/111122223333/originId/index.mpd?sessionId=session&test=123
```

Aplicação de parâmetros no DASH

A resposta do manifesto do DASH inclui os parâmetros de consulta em vários locais, incluindo segmentos de conteúdo, segmentos de anúncios e inicialização URLs. MediaTailor aplica parâmetros ao seguinte:

- Elementos de localização do manifesto DASH
- SegmentTemplate atributos de inicialização

- SegmentTemplate atributos de mídia
- Segmento de conteúdo URLs
- Segmento de anúncio URLs

Quando o cliente faz a solicitação, MediaTailor responde com um manifesto DASH semelhante ao exemplo a seguir. O primeiro período é um período de conteúdo, portanto, MediaTailor não insere o parâmetro de consulta do manifesto nele. No segundo período, que é um período de anúncio, MediaTailor insere o parâmetro de consulta do manifesto no `initialization` atributo e `media` no atributo do `SegmentTemplate` elemento. O `Location` elemento também tem os parâmetros de consulta do manifesto.

```
<?xml version="1.0" encoding="UTF-8"?>
<MPD availabilityStartTime="2018-07-27T09:48:23.634000+00:00"
  id="201" minBufferTime="PT30S" minimumUpdatePeriod="PT15S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011" publishTime="2023-02-14T23:37:43"
  suggestedPresentationDelay="PT25.000S" timeShiftBufferDepth="PT56.997S" type="dynamic"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/ittf/
PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd">
  <BaseURL>https://origin.com/contentSegments/</BaseURL>
  <Location>https://mediatailor.com/v1/dash/111122223333/originId/index.mpd?
test=123&aws.sessionId=session</Location>
  <Period duration="PT29.963S" id="28737823" start="PT143732873.178S">
    <AdaptationSet bitstreamSwitching="true" mimeType="video/mp4"
segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
      <Representation bandwidth="2200000" codecs="avc1.640029"
frameRate="30000/1001" height="540" id="1" width="960">
        <SegmentTemplate initialization="index_video_7_0_init.mp4?
m=1611174111" media="index_video_7_0_${Number$.mp4?m=1611174111"
presentationTimeOffset="4311986195351" startNumber="28737828" timescale="30000">
          <SegmentTimeline>
            <S d="180180" t="4311986911066"/>
            <S d="3003" t="4311987091246"/>
          </SegmentTimeline>
        </SegmentTemplate>
      </Representation>
    </AdaptationSet>
  </Period>
  <Period id="28737829_1" start="PT39925H48M23.141S">
```

```

    <BaseUrl>https://mediatailor.com/v1/
dashsegment/111122223333/originId/session/28737829/28737829_1/</BaseUrl>
    <AdaptationSet bitstreamSwitching="false" frameRate="30000/1001"
    mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
    subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SegmentTemplate startNumber="1" timescale="90000"/>
        <Representation bandwidth="2200000" codecs="avc1.64001f" height="540"
    id="1" width="960">
            <SegmentTemplate initialization="asset_540_2_0init.mp4?test=123"
    media="asset_540_2_0_${Number%09d$.mp4?test=123" startNumber="1" timescale="90000">
                <SegmentTimeline>
                    <S d="180180" r="6" t="0"/>
                    <S d="87087" t="1261260"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
    </AdaptationSet>
</Period>
</MPD>

```

MediaTailor Inicialização de sessão explícita HLS e DASH

AWS Elemental MediaTailor inclui os parâmetros de consulta `manifestParams` as na playlist multivariante e o rastreamento URLs na resposta quando o cliente faz uma solicitação explícita de inicialização da sessão.

Métodos de inicialização da sessão

Para inicialização explícita da sessão, você pode usar POST com corpo da solicitação ou GET com parâmetros de consulta:

1. POST com corpo de solicitação:

```

POST /v1/session/111122223333/originId/index.m3u8
{
    "adsParams": {"param1": "value1", "param2": "value2", "param3": "value3"},
    "manifestParams": {"test": "123"}
}

```

2. GET com parâmetros de consulta:

```
GET /v1/session/111122223333/originId/index.m3u8?  
ads.param1=value1&ads.param2=value2&manifestParams.test=123
```

Exemplo solicitação de inicialização de sessão

```
POST /v1/session/111122223333/originId/index.m3u8  
{  
  "adsParams": {  
    "param1": "value1",  
    "param2": "value2",  
    "param3": "value3"  
  },  
  "manifestParams": {  
    "test": "123"  
  },  
  "reportingMode": "client"  
}
```

Exemplo resposta manifesta e de rastreamento

```
{  
  "manifestUrl": "/v1/master/111122223333/originId/index.m3u8?  
aws.sessionId=session&test=123",  
  "trackingUrl": "/v1/tracking/111122223333/originId/session?test=123"  
}
```

As respostas manifestas para a sessão têm um valor específico `manifestParams` MediaTailor URLs semelhante aos fluxos de trabalho implícitos de inicialização de sessão descritos anteriormente. A principal diferença é que os parâmetros do manifesto para inicialização explícita da sessão não começam com `manifest`.

Os parâmetros de consulta do manifesto são imutáveis e você só pode defini-los na inicialização da sessão. Se um cliente fizer várias solicitações de playlist multivariantes para uma única sessão, MediaTailor não atualizará os parâmetros de consulta do manifesto após a primeira solicitação.

Fluxo de processamento de parâmetros

Você só pode especificar os parâmetros uma vez, no momento da inicialização. Os aliases de configuração são resolvidos para valores reais antes do encaminhamento. Por exemplo:

`player_params.ad_type=customized` resolve com `ad_type=abc12345` base na `ConfigurationAliases` configuração.

MediaTailor comportamento de parâmetros específicos do protocolo

AWS Elemental MediaTailor trata os parâmetros de consulta do manifesto de forma diferente para os protocolos HLS e DASH. Cada tipo de protocolo tem locais de aplicação e métodos de processamento específicos.

Comparação de tratamento de parâmetros HLS vs DASH

A tabela a seguir compara como MediaTailor manipula os parâmetros de consulta do manifesto nos protocolos HLS e DASH:

Aspecto	Comportamento do HLS	Comportamento do DASH
Aplicação de parâmetros	Aplicado diretamente ao manifesto URLs e ao segmento URLs	Aplicado a elementos, SegmentTemplate atributos e segmentos de localização URLs
Hierarquia de manifestos	Playlist multivariante → Playlists de mídia → Segmentos	MPD → Períodos → AdaptationSets → Representações
Inicialização URLs	Aplicado ao HLS init quando presente URLs	Aplicado aos SegmentTemplate atributos de inicialização
Manuseio de sessões	Parâmetros preservados nas atualizações da playlist	Parâmetros incluídos no elemento MPD Location para continuidade da sessão
Tratamento de segmentos de anúncios	Aplicado ao segmento de anúncios URLs em playlists de mídia	Aplicado aos atributos de SegmentTemplate mídia do período do anúncio

Localizações de aplicação de parâmetros

MediaTailor aplica parâmetros de consulta de manifesto aos seguintes locais:

Aplicação de parâmetros HLS

Para fluxos HLS, MediaTailor aplica parâmetros de consulta de manifesto a:

- Lista de reprodução multivariante URLs: os parâmetros são anexados às referências da lista de reprodução de mídia
- Lista de reprodução de mídia URLs: os parâmetros são incluídos no segmento URLs nas listas de reprodução de mídia
- Segmento de conteúdo URLs: todos os segmentos de conteúdo incluem os parâmetros de consulta do manifesto
- Segmento de anúncios URLs: segmentos de anúncios recebem parâmetros para roteamento e autorização de CDN
- Inicialização HLS URLs: os segmentos iniciais incluem parâmetros quando presentes no fluxo
- Segmento de ardósia URLs: o conteúdo de ardósia inclui parâmetros para um comportamento consistente da CDN

Example Exemplo de aplicação de parâmetros HLS

Dada a inicialização da sessão:

```
GET /v1/master/123456789/originId/index.m3u8?  
manifest.auth_token=abc123&manifest.region=us-west
```

A playlist multivariante inclui parâmetros nas referências da playlist de mídia:

```
#EXTM3U  
#EXT-X-VERSION:3  
#EXT-X-STREAM-INF:BANDWIDTH=2665212,RESOLUTION=960x540  
../../../../manifest/123456789/originId/session/0.m3u8?auth_token=abc123&region=us-west
```

A playlist de mídia inclui parâmetros no segmento URLs:

```
#EXTM3U  
#EXT-X-VERSION:6  
#EXT-X-TARGETDURATION:7  
#EXTINF:6.006,  
https://origin.com/segment1.ts?auth_token=abc123&region=us-west
```

```
#EXTINF:6.006,
../../../../../../../../segment/123456789/originId/session/0/2?auth_token=abc123&region=us-west
```

Aplicação de parâmetros DASH

Para fluxos do DASH, MediaTailor aplica parâmetros de consulta de manifesto a:

- Elementos de localização do MPD: o elemento Localização inclui parâmetros para solicitações de atualização do manifesto
- SegmentTemplate atributos de inicialização: o segmento URLs inicial inclui parâmetros
- SegmentTemplate atributos de mídia: os modelos de URL do segmento de mídia incluem parâmetros
- Segmento de conteúdo URLs: todos os segmentos de conteúdo gerados a partir de modelos incluem parâmetros
- Segmento do anúncio URLs: os segmentos do período do anúncio incluem parâmetros para integração com a CDN
- Redirecionamentos de relatórios do lado do servidor: 302 redirecionamentos para segmentos de anúncios preservam parâmetros

Exemplo Exemplo de aplicação de parâmetros DASH

Dada a inicialização da sessão:

```
GET /v1/dash/123456789/originId/index.mpd?
manifest.auth_token=abc123&manifest.region=us-west
```

O manifesto DASH inclui parâmetros em vários locais:

```
<MPD>
  <Location>https://mediatailor.com/v1/dash/123456789/originId/index.mpd?
auth_token=abc123&region=us-west&aws.sessionId=session</Location>
  <Period>
    <AdaptationSet>
      <Representation>
        <SegmentTemplate
          initialization="init.mp4?auth_token=abc123&region=us-west"
          media="segment_${Number$.mp4?auth_token=abc123&region=us-west"/>
        </Representation>
```

```
</AdaptationSet>  
</Period>  
</MPD>
```

MediaTailor Integração CDN e roteamento de parâmetros

AWS Elemental MediaTailor os parâmetros de consulta de manifesto permitem cenários sofisticados de integração de CDN. Você pode usá-los para roteamento dinâmico, autorização e balanceamento de carga.

Casos de uso de roteamento CDN

Os cenários comuns de integração de CDN que se beneficiam dos parâmetros de consulta do manifesto incluem o seguinte:

- Roteamento geográfico: encaminhe solicitações para MediaTailor endpoints específicos da região com base na localização do visualizador
- Autorização baseada em token: passe tokens de autorização pela CDN MediaTailor para acesso seguro ao conteúdo
- Balanceamento de carga: distribua o tráfego em vários MediaTailor endpoints usando a lógica de roteamento CDN
- Teste A/B: direcione diferentes segmentos de usuários para diferentes MediaTailor configurações de teste
- Otimização específica do dispositivo: roteie solicitações com base no tipo ou nos recursos do dispositivo

Preservação de parâmetros em camadas de CDN

MediaTailor garante que os parâmetros de consulta do manifesto sejam preservados em várias camadas de CDN e tipos de solicitação:

1. Solicitação inicial: os parâmetros são extraídos da solicitação de inicialização da sessão
2. Geração de manifesto: os parâmetros são aplicados a todos os relevantes URLs no manifesto
3. Solicitações de segmento: os parâmetros são incluídos em todos os segmentos URLs para um comportamento consistente da CDN
4. Inserção de anúncios: os parâmetros são preservados durante a inserção do anúncio e a substituição do segmento

Exemplo Fluxo de autorização CDN

O exemplo a seguir demonstra um fluxo completo de autorização de CDN usando parâmetros de consulta de manifesto:

1. O cliente solicita manifesto com token de autorização:

```
GET https://cdn.example.com/mediatailor/v1/master/123456789/originId/index.m3u8?
manifest.auth_token=jwt_token_here&manifest.user_id=12345
```

2. O CDN encaminha a solicitação para MediaTailor com parâmetros:

```
GET https://mediatailor.amazonaws.com/v1/master/123456789/originId/index.m3u8?
manifest.auth_token=jwt_token_here&manifest.user_id=12345
```

3. MediaTailor gera um manifesto com parâmetros aplicados a todos URLs:

```
#EXTM3U
#EXT-X-STREAM-INF:BANDWIDTH=2665212
../.../manifest/123456789/originId/session/0.m3u8?
auth_token=jwt_token_here&user_id=12345
```

4. As solicitações de segmentos subsequentes incluem parâmetros para autorização de CDN:

```
GET https://cdn.example.com/mediatailor/segment/123456789/originId/session/0/1?
auth_token=jwt_token_here&user_id=12345
```

Dados de relatórios e rastreamento

MediaTailor fornece duas opções para rastrear e gerar relatórios sobre a quantidade de um anúncio que um espectador assistiu. Na abordagem de relatórios de anúncios do lado do servidor, MediaTailor rastreia o anúncio e envia beacons (sinais de rastreamento) diretamente para o servidor de anúncios. Como alternativa, na abordagem de rastreamento do lado do cliente, o player do cliente (o dispositivo do usuário) rastreia o anúncio e envia os beacons para o servidor de anúncios. O tipo de relatório de anúncio usado em uma sessão de reprodução depende da solicitação específica que o player faz para iniciar a sessão. MediaTailor

Para obter informações sobre como transmitir dados da sessão e do jogador para o servidor de anúncios usando variáveis dinâmicas, consulte [MediaTailor variáveis dinâmicas de anúncios](#)

[para solicitações de ADS](#). Para obter detalhes sobre os parâmetros de inicialização da sessão, consulte [MediaTailor parâmetros de consulta do manifesto](#).

Tópicos

- [MediaTailor Rastreamento e relatórios de anúncios do lado do servidor](#)
- [Client-side rastreamento de anúncios](#)

MediaTailor Rastreamento e relatórios de anúncios do lado do servidor

AWS Elemental MediaTailor usa como padrão os relatórios do lado do servidor para rastreamento e medição abrangentes de anúncios. Com os relatórios no lado do servidor, quando o player solicita um URL de anúncio no manifesto, o serviço reporta o consumo de anúncios diretamente para o URL de rastreamento de anúncios. Depois que o player inicializa uma sessão de reprodução com MediaTailor, nenhuma outra entrada é necessária de você ou do player para realizar relatórios do lado do servidor. À medida que cada anúncio é reproduzido, MediaTailor envia beacons ao servidor de anúncios para informar quanto do anúncio foi visualizado. MediaTailor envia beacons para o início do anúncio e para a progressão do anúncio em quartis: primeiro quartil, ponto médio, terceiro quartil e conclusão do anúncio.

Para realizar a geração de relatórios de anúncios no lado do servidor

- No player, inicialize uma nova sessão de MediaTailor reprodução usando uma solicitação em um dos seguintes formatos, de acordo com seu protocolo:
 - Exemplo: formato HLS

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/<asset-id>?  
ads.<key-value-pairs-for-ads>&<key-value-pairs-for-origin-server>
```

- Exemplo: formato DASH

```
GET <mediatailorURL>/v1/dash/<hashed-account-id>/<origin-id>/<asset-id>?ads.<key-  
value-pairs-for-ads>&<key-value-pairs-for-origin-server>
```

Os pares de chave/valor são os parâmetros dinâmicos de direcionamento para rastreamento de anúncios. Para obter informações sobre como adicionar parâmetros à solicitação, consulte [the section called “Variáveis de anúncios dinâmicos”](#).

AWS Elemental MediaTailor responde à solicitação com o URL do manifesto. O manifesto contém URLs dos manifestos de mídia. Os manifestos de mídia contêm links para solicitações de segmento de anúncios.

 Note

Quando MediaTailor encontra uma barra dupla (//) em um URL de rastreamento, ela reduz as barras para uma (/).

Quando o player solicita a reprodução do URL de um segmento de anúncio (caminho /v1/segment), o AWS Elemental MediaTailor envia o beacon apropriado ao servidor por meio dos URLs de rastreamento de anúncios. Ao mesmo tempo, o serviço emite um redirecionamento para o segmento de anúncio *.ts real. O segmento de anúncios está na CloudFront distribuição da Amazon, onde MediaTailor armazena anúncios transcodificados, ou na rede de entrega de conteúdo (CDN), onde você armazenou o anúncio em cache.

As seções a seguir fornecem mais informações sobre como trabalhar com o rastreamento de anúncios do lado do servidor a partir de. MediaTailor

Tópicos

- [Server-side rastreamento com inserção de anúncios guiada pelo servidor \(SGAI\)](#)
- [Server-side glossário de faróis de rastreamento](#)
- [Server-side tempo de rastreamento e comportamento de armazenamento em cache](#)
- [MediaTailor recursos e capacidades de rastreamento do lado do servidor](#)

Server-side rastreamento com inserção de anúncios guiada pelo servidor (SGAI)

Quando você usa a inserção de anúncios guiada pelo servidor (SGAI), o rastreamento do lado do servidor usa um mecanismo de sinalização sem sessão que difere da abordagem de modo combinado descrita acima. Em vez de MediaTailor unir segmentos de anúncios ao manifesto de conteúdo (onde rastreia /v1/segment solicitações), o SGAI retorna referências de anúncios como playlists separadas em uma resposta da lista de ativos com metadados de beacon incorporados aos URIs do anúncio.

Como funciona o beaconing do lado do servidor sem sessão

As etapas a seguir descrevem como o beaconing do lado do servidor funciona nas sessões do SGAI:

1. Inicialização da sessão: o player solicita a playlist multivariante HLS com.
`aws.insertionMode=GUIDED` Server-side o relatório é o padrão (nenhum `aws.reportingMode` parâmetro é necessário). Ao contrário do modo costurado, a resposta de inicialização da sessão não inclui um `trackingUrl`
2. Manifesto armazenável em cache: MediaTailor retorna um manifesto armazenável em cache contendo `EXT-X-DATERANGE` tags `CLASS="com.apple.hls.interstitial"` e `X-ASSET-LIST` atributos apontando para o endpoint da lista de ativos MediaTailor intersticiais.
3. Lista de ativos com metadados de beacon: quando o jogador encontra uma pausa no anúncio, ele busca a lista de ativos. MediaTailor retorna uma resposta JSON em que cada URI do anúncio inclui metadados de beacon criptografados:

```
{
  "ASSETS": [
    {
      "DURATION": 30.0,
      "URI": "https://cdn.example.com/ad/master.m3u8?
awsBeaconData=<encrypted>&awsBeaconDomain=<MediaTailor-
endpoint>&awsConfigurationName=<config-name>"
    }
  ]
}
```

Quando os relatórios do lado do servidor estão ativos, a resposta não inclui uma seção. TRACKING Os URIs do anúncio carregam todos os dados do beacon.

4. Substituição da variável HLS: o player busca a playlist multivariante do anúncio. O manifesto publicitário usa `#EXT-X-DEFINE:QUERYPARAM` diretivas para transmitir os parâmetros do beacon da string de consulta do URI para os URLs do segmento por meio da substituição da variável HLS:

```
#EXTM3U
#EXT-X-DEFINE:QUERYPARAM="awsBeaconData"
#EXT-X-DEFINE:QUERYPARAM="awsBeaconDomain"
#EXT-X-DEFINE:QUERYPARAM="awsConfigurationName"
#EXTINF:5.0,
{${awsBeaconDomain}}/segment/hash/{${awsConfigurationName}}/{${awsBeaconData}}/0/0?
aws.segmentRelativePath=asset_00001.ts
```

O player resolve as `{${awsConfigurationName}}` variáveis `{${awsBeaconData}}`, `{${awsBeaconDomain}}`, e usando os valores da string de consulta

do URI do manifesto do anúncio e, em seguida, solicita cada segmento do anúncio por meio MediaTailor de.

5. Sinalizador acionado por solicitação de segmento: à medida que o player solicita cada segmento de anúncio, a solicitação é encaminhada. MediaTailor O serviço decifra os dados do beacon, determina a posição do segmento no anúncio (impressão, primeiro quartil, ponto médio, terceiro quartil ou completo) e dispara o farol de rastreamento VAST apropriado para o servidor de anúncios. MediaTailor em seguida, redireciona o player para o segmento real do conteúdo do anúncio.

Requisitos do jogador para beaconing do lado do servidor SGAI

Para usar o beaconing do lado do servidor com o SGAI, seu player deve atender aos seguintes requisitos:

- HLS versão 11 ou posterior
- Support for EXT-X-DATERANGE with CLASS attribute for HLS Interstitials
- Support para substituição de #EXT-X-DEFINE:QUERYPARAM variáveis (RFC 8216bis). O player deve decodificar em porcentagem os valores dos parâmetros de consulta antes de substituí-los em URLs de segmentos.

Note

Atualmente, o beaconing do lado do servidor SGAI é compatível somente com HLS. O DASH ainda não é compatível com o beaconing do lado do servidor SGAI.

Comparação com o rastreamento do lado do servidor em modo combinado

A tabela a seguir resume como o rastreamento do lado do servidor difere entre a inserção de anúncios agrupados e a inserção guiada pelo servidor:

Aspecto	Costurado (SSAI)	Server-guided (SGAI)
Capacidade de armazenamento em cache do manifesto	Per-session, não pode ser armazenado em cache	Armazenável em cache, compartilhado entre os espectadores

Aspecto	Costurado (SSAI)	Server-guided (SGAI)
Roteamento de segmentos de anúncios	Por meio do <code>/v1/segment/</code> uso do ID da sessão	<code>/v1/segment/</code> Usando um blob de dados de beacon criptografado
Estado da sessão para beacons	Armazenado por sessão em MediaTailor	Sem sessão — todo o estado é transmitido no parâmetro criptografado <code>awsBeaconData</code>
URL de rastreamento no início da sessão	Retornado na resposta de inicialização da sessão	Não fornecido — os dados do beacon são incorporados em URIs de anúncios em cada resposta da lista de ativos
Suporte para DASH	Compatível	Sem suporte no momento

Note

Para sessões ao vivo do SGAI, você pode ativar a pré-busca de anúncios baseada em manifestos usando `aws.guidedPrefetchMode=MANIFEST`. Isso é separado da API de pré-busca baseada em agendamento usada com sessões agrupadas (SSAI). Para obter detalhes, consulte [Pré-busca guiada com batimento cardíaco manifesto](#).

Server-side glossário de faróis de rastreamento

MediaTailor o rastreamento do lado do servidor usa um conjunto padronizado de beacons para relatar o progresso da visualização de anúncios aos servidores de anúncios e serviços de verificação. Esses beacons se alinham aos padrões do Interactive Advertising Bureau (IAB) para medição de anúncios em vídeo e fornecem relatórios precisos sobre impressões de anúncios e taxas de conclusão.

Server-side tipos de faróis de rastreamento

Tipo de farol	Quando demitido	Finalidade	Detalhes do tempo
Impressão	Quando o player solicita o primeiro segmento de anúncio	Indica que o conteúdo do anúncio começou a ser carregado e está prestes a ser exibido para o espectador	Demitido na primeira /v1/ segment solicitação de um anúncio. Está alinhado às diretrizes do IAB que exigem que o conteúdo do anúncio comece a ser carregado antes de contar uma impressão. Veja Server-side fluxo de trabalho de rastreamento a sequência completa.
Início	Quando o player começa a renderizar o conteúdo do anúncio	Confirma que a reprodução do anúncio foi realmente iniciada	Normalmente é acionado simultaneamente com o farol de impressão na primeira solicitação de segmento, mas representa o início real da renderização do anúncio. Essa distinção é importante e para serviços de verificação que rastreiam os eventos de impressão e de início separadamente.
Primeiro quartil	Quando o jogador atinge 25% da duração do anúncio	Mede a visualização contínua do anúncio durante o primeiro trimestre do anúncio	Acionado quando o jogador solicita o segmento que contém o ponto de 25% da duração do anúncio. Por exemplo, em um anúncio de 20 segundos com segmentos de 2 segundos, isso normalmente seria acionado na solicitação do terceiro segmento (aproxima

Tipo de farol	Quando demitido	Finalidade	Detalhes do tempo
			damente 4 a 6 segundos após o início do anúncio).
Ponto médio	Quando o jogador atinge 50% da duração do anúncio	Mede a visualização contínua do anúncio em metade do anúncio	Acionado quando o jogador solicita o segmento que contém o ponto de 50% da duração do anúncio. Por exemplo, em um anúncio de 20 segundos com segmentos de 2 segundos, isso normalmente seria acionado na solicitação do 5º segmento (aproximadamente 8 a 10 segundos após o início do anúncio).
Terceiro quartil	Quando o jogador atinge 75% da duração do anúncio	Mede a visualização contínua de anúncios em três quartos do anúncio	Acionado quando o jogador solicita o segmento que contém o ponto de 75% da duração do anúncio. Por exemplo, em um anúncio de 20 segundos com segmentos de 2 segundos, isso normalmente seria acionado na solicitação do 8º segmento (aproximadamente 14 a 16 segundos após o início do anúncio).

Tipo de farol	Quando demitido	Finalidade	Detalhes do tempo
Concluído	Quando o jogador chega ao final do anúncio	Confirma que todo o anúncio foi entregue ao espectador	Acionado quando o jogador solicita o segmento final do anúncio. Isso indica que o espectador potencialmente viu todo o conteúdo do anúncio. Por exemplo, em um anúncio de 20 segundos com segmentos de 2 segundos, isso normalmente seria acionado na solicitação do 10º segmento (aproximadamente 18 a 20 segundos após o início do anúncio).

Note

O momento exato do disparo do farol depende da duração do segmento e da duração do anúncio. MediaTailor calcula a solicitação de segmento apropriada que corresponde a cada posição do quartil com base na duração específica do anúncio e na estrutura do segmento.

Server-side tempo de rastreamento e comportamento de armazenamento em cache

Nos relatórios do lado do servidor, MediaTailor aciona eventos de rastreamento com base nas solicitações reais do segmento do player, não na análise do manifesto ou nas atividades de pré-carregamento. Essa abordagem garante uma contagem precisa de impressões que se alinha aos padrões do setor para medição de anúncios em vídeo.

Princípios-chave de temporização

MediaTailor o rastreamento do lado do servidor segue esses princípios fundamentais de temporização:

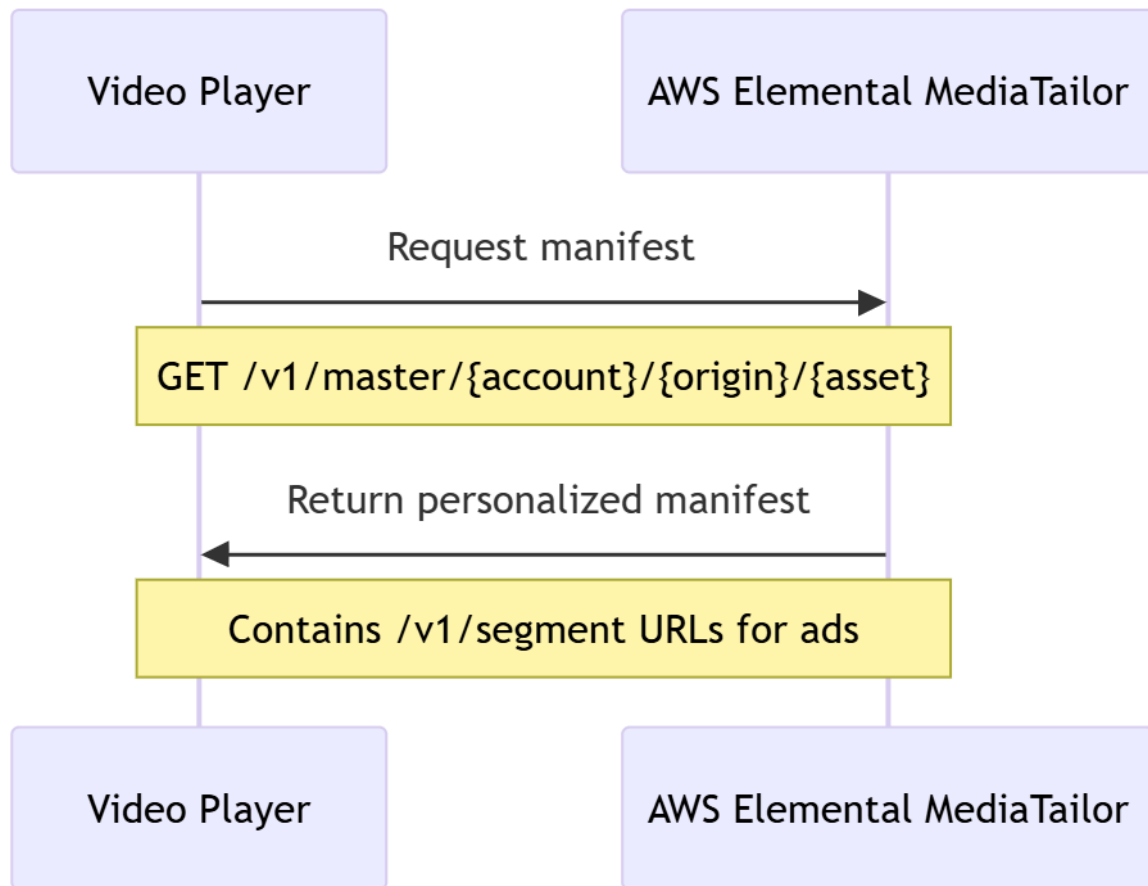
- Os eventos de rastreamento são acionados em solicitações reais de segmentos - os beacons são enviados somente quando o player faz solicitações HTTP para /v1/segment URLs, não durante a análise ou o armazenamento em cache do manifesto.
- O armazenamento em cache e o pré-carregamento de manifestos pelo jogador NÃO acionam eventos - Os jogadores podem analisar, armazenar em cache ou pré-carregar informações do manifesto sem gerar nenhum evento de rastreamento.
- A pré-busca de segmentos acionará eventos. Se os jogadores pré-buscarem segmentos de anúncios reais antes da reprodução, isso segue o comportamento padrão do setor, em que as solicitações de segmentos constituem impressões válidas.
- Cada v1/segment /request aciona o sinal apropriado: o evento de rastreamento específico (impressão, quartil, conclusão) é determinado pela posição do anúncio e pelo segmento que está sendo solicitado.
- O tempo está alinhado aos padrões do IAB — a abordagem segue as diretrizes do Interactive Advertising Bureau para medição de anúncios em vídeo e contagem de impressões.

Server-side fluxo de trabalho de rastreamento

Os diagramas a seguir ilustram o fluxo de trabalho completo de rastreamento do lado do servidor, mostrando quando os eventos de rastreamento são acionados em relação às solicitações dos jogadores:

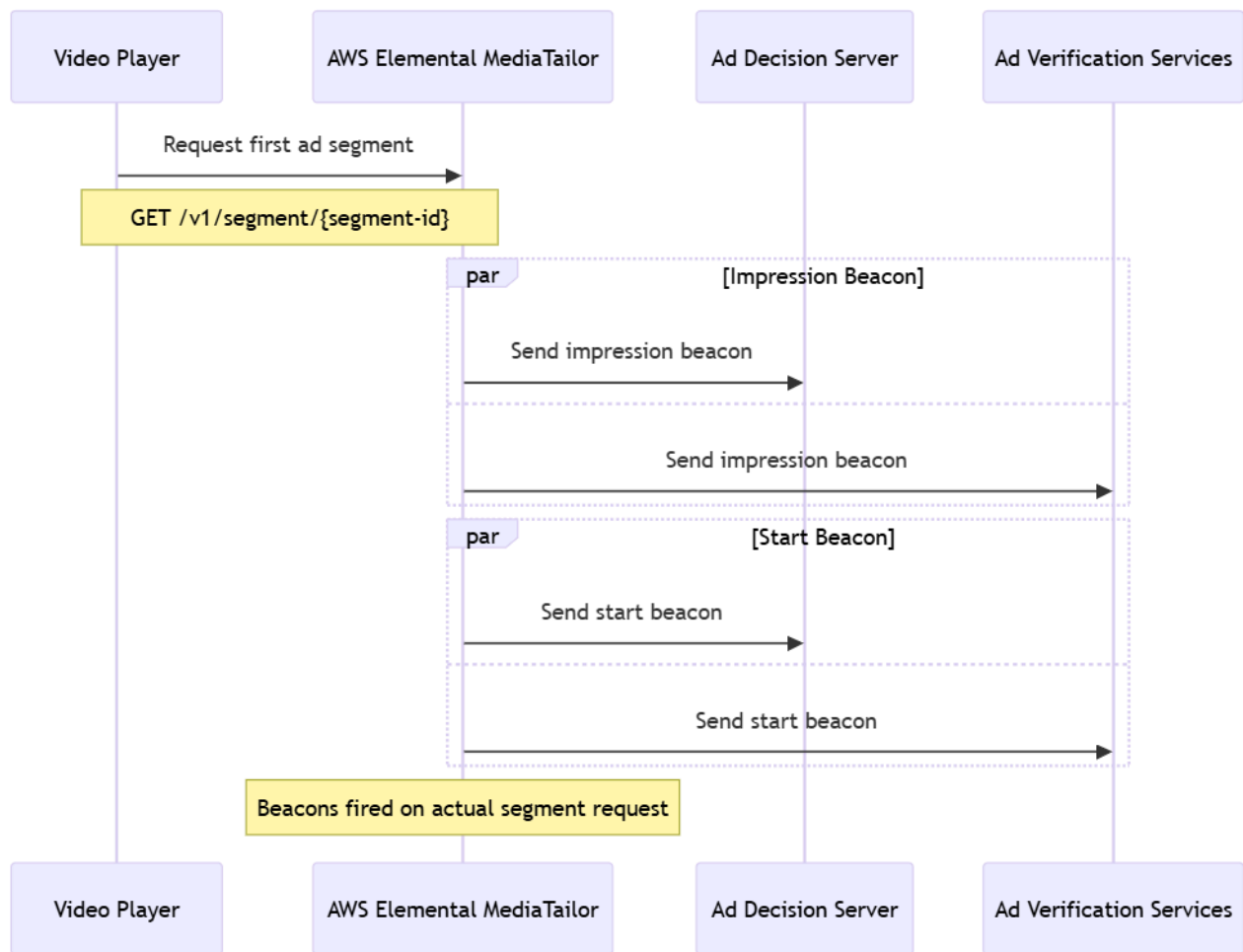
Fase 1: Inicialização da sessão

O player solicita um manifesto de MediaTailor, que retorna um manifesto personalizado contendo URLs de segmentos de anúncios:



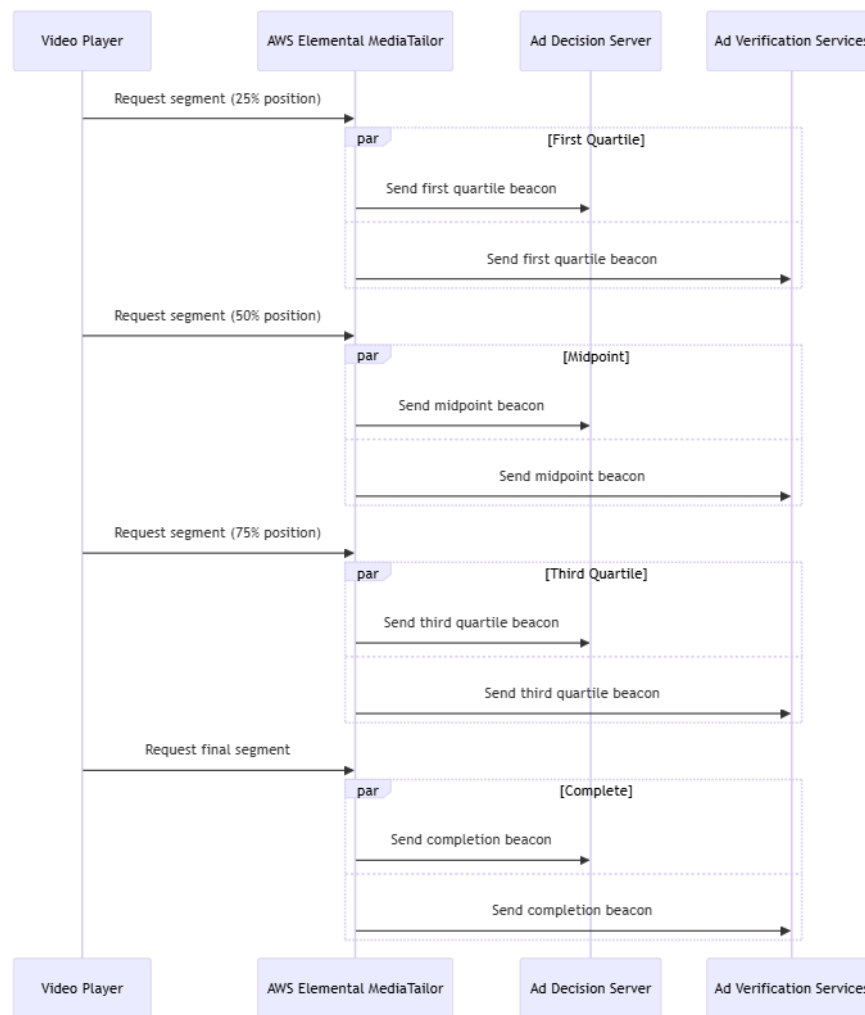
Fase 2: Rastreamento de solicitações de anúncios e impressões

Quando o player solicita o primeiro segmento do anúncio, MediaTailor dispara os beacons de impressão e de início para o Ad Decision Server e para os Ad Verification Services:



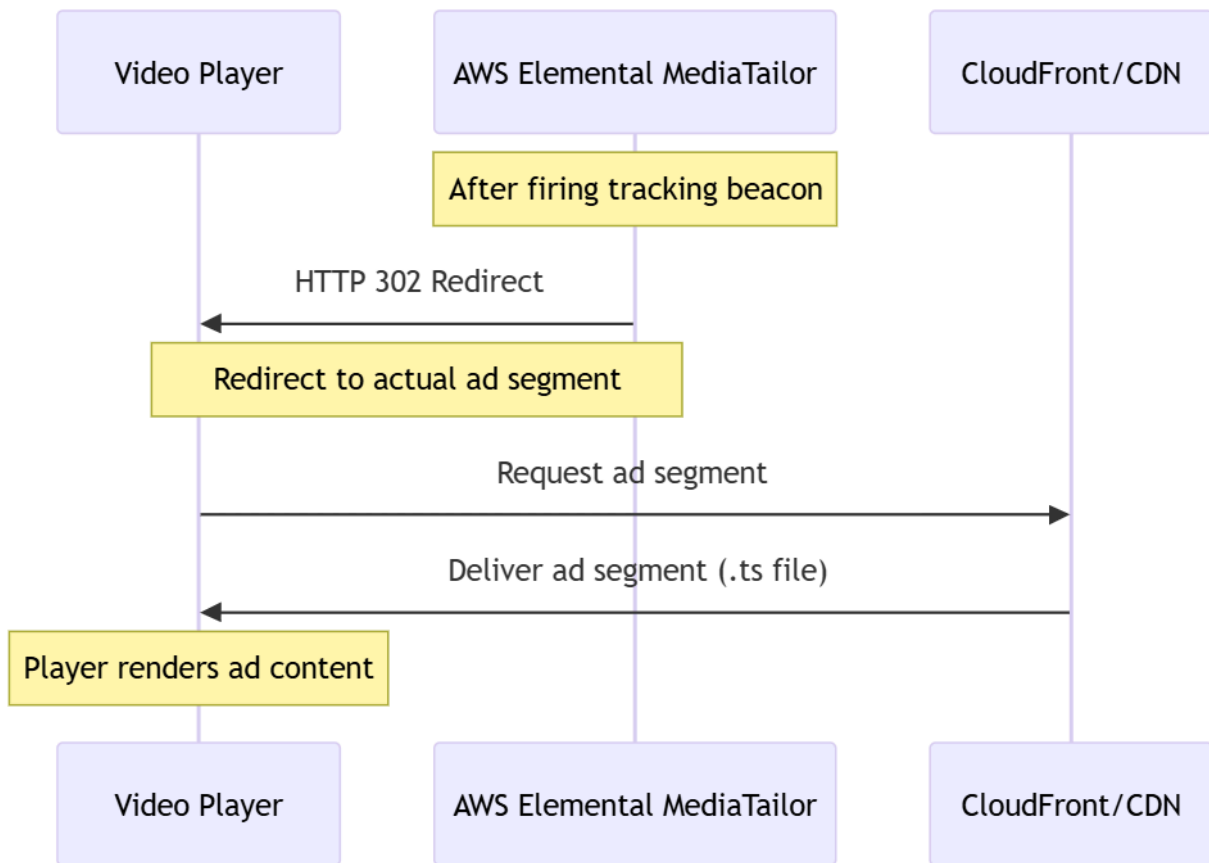
Fase 3: Rastreamento de quartis

MediaTailor dispara faróis de quartil (primeiro quartil, ponto médio, terceiro quartil, conclusão) com base nas solicitações subsequentes do segmento:



Fase 4: Entrega por segmento

Depois de disparar os beacons de rastreamento, MediaTailor redireciona para o segmento de anúncio real da Amazon ou do seu CDN: CloudFront



O fluxo de trabalho de rastreamento do lado do servidor inclui os seguintes principais comportamentos de temporização:

1. **Inicialização da sessão** - O jogador solicita um manifesto de MediaTailor. MediaTailor retorna um manifesto personalizado contendo URLs de segmentos de anúncios com o `/v1/segment` caminho.
2. **Análise e armazenamento em cache do manifesto** - O player analisa o manifesto e pode pré-carregar ou armazenar em cache as informações do segmento. Nenhum evento de rastreamento é acionado durante essa fase, independentemente do comportamento de cache do jogador.
3. **Solicitação de segmento de anúncio e rastreamento de impressões**: quando o player realmente solicita o primeiro segmento de anúncio (normalmente para reprodução), MediaTailor dispara o sinal de impressão e inicia o rastreamento do evento no Ad Decision Server e nos Serviços de

Verificação de Anúncios. Isso ocorre na solicitação HTTP real para o `/v1/segment` URL, não quando o manifesto é analisado.

4. Rastreamento de quartil com base nas solicitações do segmento: MediaTailor dispara beacons de quartil (primeiro quartil, ponto médio, terceiro quartil, conclusão) para o Ad Decision Server e para os Ad Verification Services com base nas solicitações subsequentes do segmento que correspondem às posições calculadas do quartil dentro da duração do anúncio.
5. Entrega por segmento - Depois de acionar o sinalizador de rastreamento apropriado, MediaTailor emite um redirecionamento HTTP para o segmento real do anúncio (da Amazon CloudFront ou da sua CDN).

Considerações sobre armazenamento em cache e pré-carregamento do player

MediaTailor o rastreamento do lado do servidor foi projetado para ser compatível com várias estratégias de armazenamento em cache e pré-carregamento do player, mantendo a medição precisa das impressões:

- Pré-carregamento do manifesto - Jogadores que pré-carregam ou armazenam em cache as informações do manifesto não acionam eventos de rastreamento. Os eventos de rastreamento só são acionados quando solicitações reais de segmento são feitas.
- Pré-busca de segmentos - Se um player pré-buscar segmentos de anúncios antes da reprodução, os eventos de rastreamento serão acionados quando esses segmentos forem solicitados, potencialmente antes do tempo real de reprodução. Esse comportamento está alinhado aos padrões do setor que consideram as solicitações de segmentos como impressões válidas.
- Armazenamento em buffer do player - O comportamento padrão do buffer do player (solicitando segmentos um pouco antes da reprodução) acionará eventos de rastreamento nos momentos apropriados, com base no padrão de solicitação do segmento.

Solução de problemas de discrepâncias de rastreamento

Se você observar discrepâncias entre o rastreamento do MediaTailor lado do servidor e as métricas de terceiros, considere os seguintes fatores:

- Diferenças de comportamento do jogador - jogadores diferentes podem ter estratégias variadas de pré-busca e armazenamento em buffer que afetam o momento em que as solicitações de segmentos são feitas.

- Condições de rede - Condições de rede ruins podem fazer com que os jogadores solicitem segmentos várias vezes ou em intervalos diferentes do esperado.
- Configuração de CDN - O armazenamento incorreto de /v1/segment solicitações na CDN pode levar a eventos de rastreamento perdidos ou duplicados.
- Gerenciamento de sessão - Certifique-se de que cada sessão de reprodução use um identificador de sessão exclusivo para evitar conflitos de eventos de rastreamento.

Para obter orientações detalhadas sobre solução de problemas, consulte [Solução de problemas comuns do](#).

MediaTailor recursos e capacidades de rastreamento do lado do servidor

AWS Elemental MediaTailor aplica automaticamente esses recursos integrados de rastreamento do lado do servidor para otimizar a precisão e a confiabilidade da medição de anúncios. O sistema evita a duplicação de beacons, gerencia o tráfego durante períodos de alto volume, mantém o sequenciamento adequado de eventos e fornece monitoramento abrangente do desempenho sem exigir nenhuma configuração de sua parte. Você só precisa garantir que seu servidor de decisão de anúncio (ADS) forneça os beacons de rastreamento na resposta do VAST.

Note

Esses recursos estão disponíveis para novos clientes a partir de 30 de setembro de 2025. Os clientes existentes terão acesso ao longo de 2025 como parte das melhorias contínuas do serviço. Se você quiser acesso imediato a esses recursos, entre em contato com o [AWS Support](#).

Note

Esses recursos se aplicam aos métodos de inserção de anúncios costurados (SSAI) e guiados pelo servidor (SGAI). Os tipos de farol e o tempo são os mesmos nos dois modos. Eles diferem na forma como MediaTailor acionam os beacons — consulte [Server-side rastreamento com inserção de anúncios guiada pelo servidor \(SGAI\)](#) para obter detalhes sobre os beacons do lado do servidor SGA.

Desduplicação de beacons

MediaTailor impede o disparo de faróis duplicados para eventos publicitários idênticos. O sistema de rastreamento do lado do servidor envia cada sinal de impressão, quartil e conclusão somente uma vez por sessão de visualização do anúncio. Quando os players de vídeo solicitam o mesmo segmento de anúncio várias vezes devido às condições da rede, alterações na taxa de bits ou estratégias de buffer, MediaTailor rastreia os beacons disparados e bloqueia transmissões redundantes.

A desduplicação resolve automaticamente cenários comuns que causam um aumento no número de beacons:

- Streaming de taxa de bits adaptável: quando os jogadores baixam diferentes variantes de qualidade do mesmo segmento de anúncio
- Cenários de repetição de rede - Quando os jogadores solicitam segmentos novamente devido a problemas de rede ou tempos limite
- Estratégias de buffer de jogadores - Quando os jogadores pré-buscam ou rebuscam segmentos para fins de buffer

O sistema foi projetado para disparar faróis de impressão apenas uma vez, mesmo quando os jogadores alternam entre diferentes níveis de qualidade.

Limitação adaptativa e novas tentativas de beacon

MediaTailor gerencia automaticamente as taxas de tráfego do beacon com base nos indicadores de resposta do servidor. O sistema monitora os padrões de resposta HTTP, os tempos limite de conexão e os códigos de erro para detectar o congestionamento e, em seguida, ajusta as taxas de tráfego adequadamente. Quando o sistema identifica indicadores de estresse do servidor, ele reduz as taxas de tráfego para o domínio afetado e aumenta automaticamente as taxas quando os servidores demonstram capacidade aprimorada.

O sistema monitora a integridade do servidor usando estes indicadores:

- Tempos limite de conexão HTTP - Quando as plataformas de medição não respondem dentro dos prazos esperados
- Códigos de resposta de erro - respostas 503, 504 e 507 que indicam sobrecarga do servidor. Seu servidor de anúncios também deve oferecer suporte a esses códigos de erro para obter compatibilidade total.

- Padrões de resposta - mudanças no desempenho da plataforma de medição que indicam problemas de capacidade

O comportamento de repetição tenta automaticamente a entrega por até 1 hora, com atrasos mínimos de 30 segundos entre as tentativas. Esse comportamento de nova tentativa não pode ser configurado.

Gerenciamento de tráfego de beacon por segundo

Você pode definir limites de TPS para controlar as taxas de entrega de beacons. Essa é a única configuração configurável para recursos de rastreamento do lado do servidor. Account-level os limites limitam o número total de solicitações de rastreamento de anúncios enviadas por todos os parceiros de medição. MediaTailor impõe um limite mínimo de TPS de 10.000 para fornecer capacidade suficiente para operações em escala empresarial.

Envie um ticket de AWS suporte para estabelecer limites de TPS com as seguintes informações:

- ID da conta da AWS — Seu identificador de conta específico
- Região de destino - A região da AWS em que você deseja aplicar o limite de TPS
- Limite de TPS desejado - Seu limite de transações necessárias por segundo (mínimo 10.000)

Por padrão, não há limite de TPS. Você pode solicitar um limite de TPS se seu servidor de decisão de anúncios (ADS) exigir, mas o limite deve ser maior que 10.000 TPS. MediaTailor não excederá o limite especificado, mas não garantirá uma taxa de transferência consistente até esse limite. Seu servidor de decisão de anúncios informará quais limites de TPS ele pode suportar.

In-order balizamento

MediaTailor mantém automaticamente a entrega sequencial de eventos de rastreamento de anúncios. O sistema preserva a ordem dos beacons mesmo quando ocorrem problemas de rede, novas tentativas ou gerenciamento de tráfego. Isso garante que os parceiros de medição recebam os eventos na ordem correta para análises precisas.

O sistema segue a sequência de faróis padrão da indústria:

1. Iniciar eventos - Ação quando a reprodução do anúncio começar
2. Eventos do primeiro quartil - Disparem com 25% da conclusão
3. Eventos de ponto médio - Atire a 50% da conclusão

4. Eventos do terceiro quartil - Atire com 75% da conclusão
5. Eventos de conclusão - Ação quando os anúncios terminarem

Esses recursos funcionam juntos automaticamente:

- Os faróis são mantidos durante a aceleração para manter a ordem correta
- Cada domínio do parceiro de medição tem filas de eventos separadas para evitar interrupções durante os ajustes de taxa
- A deduplicação rastreia o tipo de evento e a posição do cronograma, mantendo a ordem cronológica

Client-side rastreamento de anúncios

Usando a API de rastreamento AWS Elemental MediaTailor do lado do cliente, você pode incorporar controles de player durante intervalos publicitários em fluxos de trabalho de streaming. No rastreamento do lado do cliente, o player ou cliente emite eventos de rastreamento, como impressão e quartil de anúncios, para o Ad Decision Server (ADS) e outras entidades de verificação de anúncios. Esses eventos monitoram o status geral do intervalo publicitário e a disponibilidade individual do anúncio em cada intervalo. Para obter mais informações sobre impressão e quartil (ADS) e outras entidades de verificação de anúncios. Para obter mais informações sobre impressão e quartil e balizamento de anúncios, consulte [Client-side balizamento](#). Para obter mais informações sobre o ADS e outras entidades de verificação de anúncios, consulte [Client-side integrações de rastreamento de anúncios](#)

Para obter informações sobre como passar os parâmetros do jogador e os dados da sessão para o ADS para rastreamento do lado do cliente, consulte e [MediaTailor variáveis de jogador para solicitações de ADS](#) [MediaTailor variáveis de sessão para solicitações de ADS](#)

Client-side o rastreamento permite funcionalidades como as seguintes:

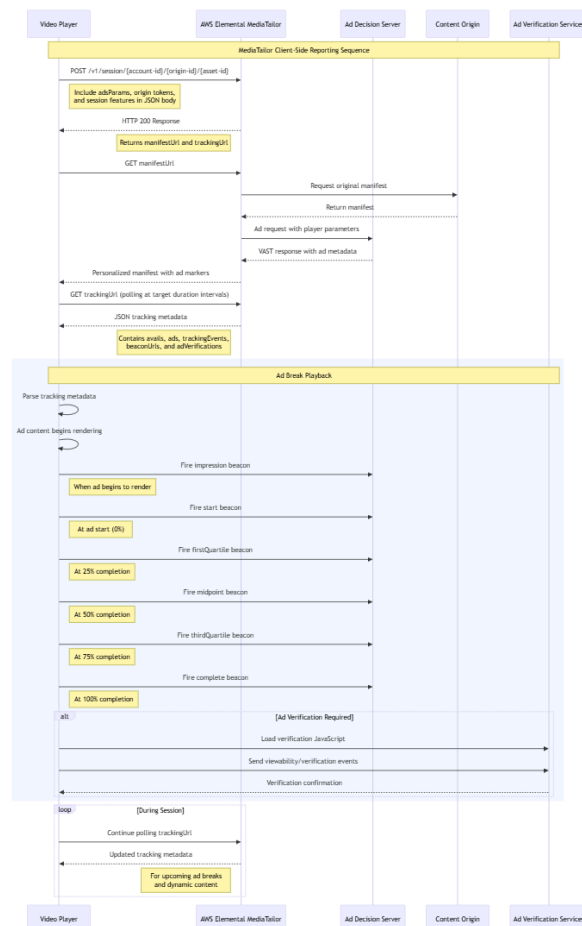
- Ad-break cronômetros de contagem regressiva - Para obter mais informações, consulte [Adicionar cronômetro de contagem regressiva](#)
- Clique no anúncio: para obter mais informações, consulte [Clique no anúncio](#)
- Exibição de anúncios complementares - Para obter mais informações, consulte [Anúncios complementares](#).
- Anúncios ignoráveis - Para obter mais informações, consulte [Anúncios ignoráveis](#)

- Exibição de ícones VAST para conformidade com a privacidade - Para obter mais informações, consulte [Ícones para o Google Por que esse anúncio \(WTA\)](#).
- Controle da depuração do jogador durante anúncios - Para obter mais informações, consulte. [Esfregando](#)

Usando a API de rastreamento MediaTailor do lado do cliente, você pode enviar metadados para o dispositivo de reprodução que habilita a funcionalidade além do rastreamento do lado do cliente:

Client-side fluxo de trabalho de relatórios

O diagrama a seguir mostra o fluxo de trabalho completo de relatórios do lado do cliente, desde a inicialização da sessão até a reprodução de anúncios e o beaconing:



O fluxo de trabalho de relatórios do lado do cliente inclui as seguintes etapas:

1. Inicialização da sessão - O player de vídeo envia uma solicitação POST ao endpoint da MediaTailor sessão com metadados JSONadsParams, incluindo tokens de origem e recursos da sessão. MediaTailor responde com `manifestUrl` e `trackingUrl` para a sessão.

2. Solicitação de manifesto e decisão de anúncio - O jogador solicita o manifesto personalizado de MediaTailor. MediaTailor solicita o manifesto do conteúdo original da origem, faz uma solicitação de anúncio ao Ad Decision Server (ADS) usando parâmetros do player, recebe uma resposta VAST com metadados do anúncio e entrega um manifesto personalizado com marcadores de anúncios para o player.
3. Recuperação de dados de rastreamento - O jogador pesquisa o URL de rastreamento em intervalos regulares (duração desejada correspondente para HLS ou período mínimo de atualização para DASH). MediaTailor retorna metadados de rastreamento JSON contendo disponibilidades, anúncios, eventos de rastreamento, URLs de beacon e dados de verificação de anúncios.
4. Reprodução e sinalização de anúncios: durante os intervalos publicitários, o player analisa os metadados de rastreamento, dispara beacons de impressão quando os anúncios começam a ser renderizados, dispara beacons de quartil (início, primeiro quartil, ponto médio, terceiro quartil, completo) no momento apropriado, carrega e executa a verificação de anúncios, se necessário, e envia eventos para serviços de verificação terceirizados. JavaScript viewability/verification
5. Pesquisa contínua - O player continua pesquisando o URL de rastreamento durante toda a sessão para receber metadados atualizados para os próximos anúncios e conteúdo dinâmico.

Esse fluxo de trabalho permite recursos avançados, como cronômetros de contagem regressiva de anúncios, funcionalidade de clique, anúncios complementares, anúncios ignoráveis e exibição de ícones VAST para conformidade com a privacidade.

Tópicos

- [Habilitando o rastreamento do lado do cliente](#)
- [Parâmetros do servidor de anúncios](#)
- [Parâmetros de consulta de interação de origem](#)
- [Session-configured features](#)
- [Melhores práticas para rastreamento do lado do cliente](#)
- [Client-side esquema e propriedades de acompanhamento de anúncios](#)
- [Ad-tracking tempo de atividade](#)
- [Controles e funcionalidades do player para rastreamento de anúncios do lado do cliente](#)
- [Client-side balizamento](#)
- [Modo híbrido com beacons de anúncios do lado do servidor](#)

- [Client-side integrações de rastreamento de anúncios](#)
- [Paginando beacons de anúncios com GetTracking](#)

Habilitando o rastreamento do lado do cliente

Você ativa o rastreamento do lado do cliente para cada sessão. O player cria um HTTP POST para o endpoint do MediaTailor prefixo de inicialização da sessão da configuração. Opcionalmente, o player pode enviar metadados adicionais para usar MediaTailor ao fazer chamadas publicitárias, chamar a origem de um manifesto e invocar ou desativar MediaTailor recursos no nível da sessão.

O exemplo a seguir mostra a estrutura dos metadados JSON:

```
{
  "adsParams": {
    "param1": "value1",
    "param2": "value2",
    "origin_access_token": "abc123",
    "overlayAvails": "on"
  },
  "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.

Use o MediaTailor console ou a API para configurar a URL do modelo de solicitação do ADS para referenciar esses parâmetros. No exemplo a seguir, `player_params.param1` são os parâmetros do player `param1`, e `player_params.param2` são os parâmetros do player `param2`.

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

Parâmetros do servidor de anúncios

No nível mais alto da estrutura JSON está um objeto `adsParams` JSON. Dentro desse objeto, há `key/value` pares que MediaTailor podem ser lidos e enviados ao servidor de anúncios em todas as solicitações de sessão. MediaTailor suporta os seguintes servidores de anúncios:

- Gerenciador de anúncios do Google
- SpringServe
- FreeWheel

- Publica

Parâmetros de consulta de interação de origem

Quaisquer key/value pares reservados no nível mais alto da estrutura JSON, como, e adsParams availSuppressionoverlayAvails, não são adicionados ao URL da solicitação de origem na forma de parâmetros de consulta. Cada solicitação de manifesto de MediaTailor sessão feita à origem contém esses parâmetros de consulta. A origem ignora parâmetros de consulta estranhos. Por exemplo, MediaTailor pode usar os key/value pares para enviar tokens de acesso à origem.

Session-configured features

Use a estrutura JSON de inicialização de sessão para ativar, desativar ou substituir MediaTailor recursos como, e. overlayAvails availSuppression adSignaling Qualquer configuração de recurso passada durante a inicialização da sessão substitui a configuração no nível da configuração. MediaTailor

Note

Os metadados enviados MediaTailor na inicialização da sessão são imutáveis, e metadados adicionais não podem ser adicionados durante a sessão. Use SCTE-35 marcadores para transmitir dados que mudam durante a sessão. Para obter mais informações, consulte [MediaTailor variáveis de sessão para solicitações de ADS](#).

Example: Realizando o rastreamento de anúncios do lado do cliente para 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: Executando o rastreamento de anúncios do lado do cliente para o DASH

```
POST mediatailorURL/v1/session/hashed-account-id/origin-id/asset-id.mpd
```

```
{
  "adsParams": {
    "deviceType": "androidmobile",
    "uid": "xjhddli-9189901-uic"
  }
}
```

Parâmetro do modo de relatório

Você pode especificar o modo de relatório ao inicializar uma sessão incluindo o `reportingMode` parâmetro no corpo da solicitação. Esse parâmetro controla se MediaTailor realiza o rastreamento de anúncios do lado do cliente ou do lado do servidor para a sessão.

- **client**- O player realiza o rastreamento de anúncios e envia beacons para o servidor de anúncios. Esse é o modo padrão se não `reportingMode` for especificado.
- **server**- MediaTailor realiza o rastreamento de anúncios do lado do servidor e envia beacons diretamente para o servidor de anúncios.

Exemplo Inicialização da sessão com modo de relatório do lado do servidor

POST *mediatailorURL*/v1/session/*hashed-account-id/origin-id/asset-id*.m3u8

```
{
  "adsParams": {
    "deviceType": "ipad",
    "uid": "abdgfdyei-2283004-ueu"
  },
  "reportingMode": "server"
}
```

Exemplo Inicialização da sessão com modo de relatório do lado do cliente (explícito)

POST *mediatailorURL*/v1/session/*hashed-account-id/origin-id/asset-id*.mpd

```
{
  "adsParams": {
    "deviceType": "androidmobile",
    "uid": "xjhddli-9189901-uic"
  },
  "reportingMode": "client"
}
```

```
}
```

Note

O `reportingMode` parâmetro é definido na inicialização da sessão e não pode ser alterado durante a sessão. Se não `reportingMode` for especificado, o MediaTailor padrão é o relatório do lado do cliente para manter a compatibilidade com versões anteriores.

Uma resposta bem-sucedida é um HTTP 200 com um corpo de resposta. O corpo contém um objeto JSON com uma `manifestUrl` e uma `trackingUrl` chave. Os valores são URLs relativos que o player pode usar para fins de reprodução e rastreamento de eventos publicitários.

```
{
  "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"
}
```

Para obter mais informações sobre o esquema de rastreamento do lado do cliente, consulte [Client-side esquema e propriedades de acompanhamento de anúncios](#)

Melhores práticas para rastreamento do lado do cliente

Esta seção descreve as melhores práticas para rastreamento do lado do cliente em MediaTailor fluxos de trabalho ao vivo e VOD.

Fluxos de trabalho ao vivo

Pesquise o endpoint de rastreamento em um intervalo que corresponda a cada duração-alvo para o HLS ou o período mínimo de atualização para o DASH, para ter sempre os metadados de rastreamento de anúncios mais atuais. Combinar esse intervalo é especialmente importante em fluxos de trabalho em que os criativos podem ter um componente interativo ou de sobreposição.

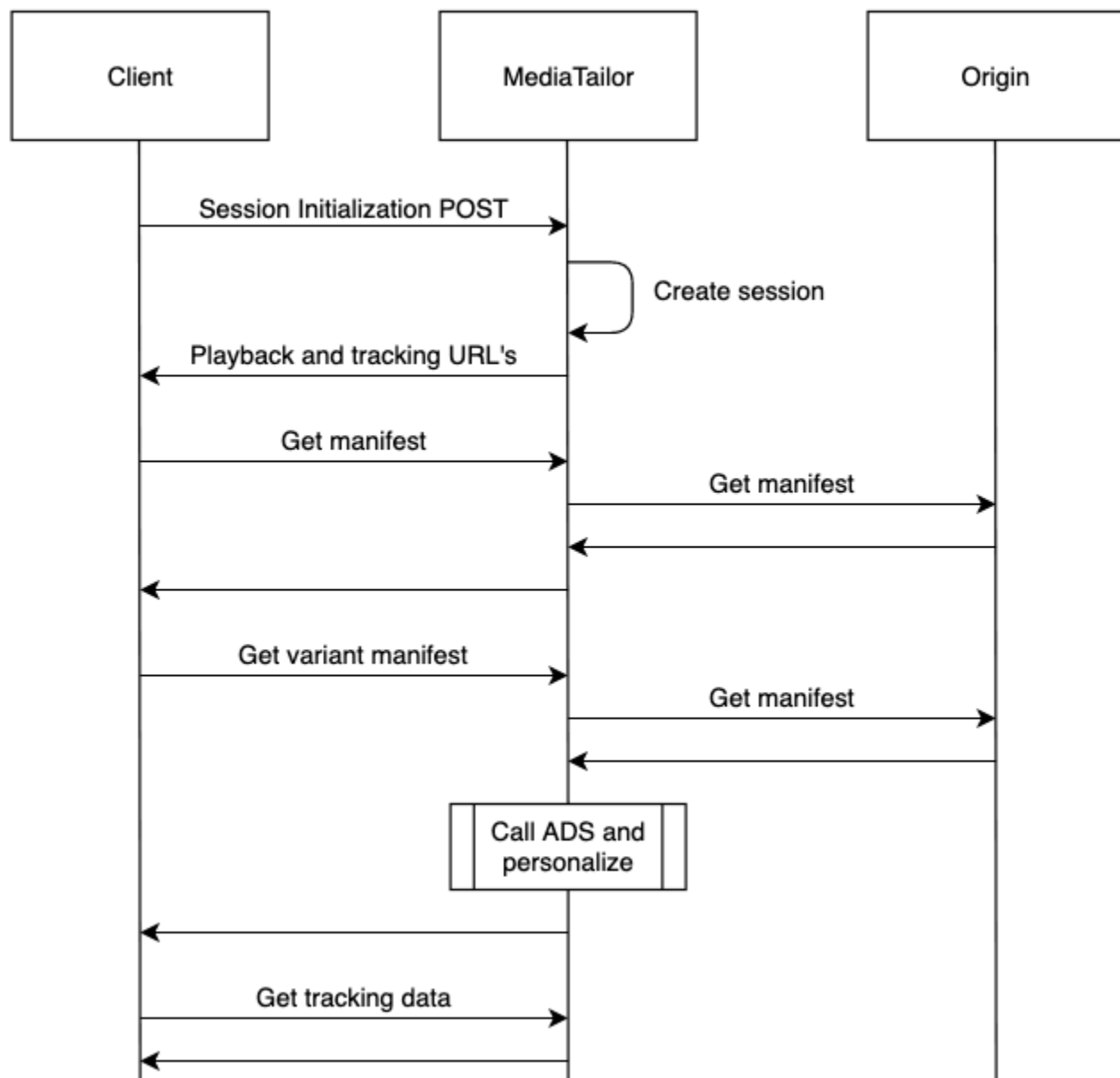
Note

Alguns jogadores apoiam os ouvintes do evento, o que pode ser usado como uma alternativa à votação. Por exemplo, o recurso de decoração do ID de MediaTailor anúncio precisaria ser ativado para cada sessão. Para obter mais informações, consulte [Decoração de ID](#)

[do anúncio](#). O uso desse recurso coloca um identificador de intervalo de datas (HLS) ou elemento de evento (DASH) sobre cada anúncio disponível. Os jogadores podem usar essas tags de manifesto como uma solicitação para chamar o endpoint de MediaTailor rastreamento da sessão.

Fluxos de trabalho de VOD

Após a inicialização bem-sucedida da sessão e depois de MediaTailor receber o primeiro manifesto contendo mídia, você só precisa chamar o endpoint de rastreamento uma vez.



Server-guided inserção de anúncios

Server-guided As sessões de inserção de anúncios (SGAI) não usam a GetTracking API. Em vez disso, quando você usa `aws.reportingMode=CLIENT`, MediaTailor fornece informações de rastreamento na TRACKING seção de cada resposta da lista de ativos quando os jogadores solicitam conteúdo publicitário. A resposta de inicialização da sessão não inclui um `trackingUrl`.

A resposta da lista de ativos para sessões SGAI rastreadas pelo lado do cliente tem a seguinte estrutura:

```
{
  "ASSETS": [
    { "DURATION": 20.0, "URI": "https://cdn.example.com/ad1/master.m3u8" },
    { "DURATION": 10.0, "URI": "https://cdn.example.com/ad2/master.m3u8" }
  ],
  "TRACKING": {
    ...VAST tracking events and beacon URLs for each ad...
  }
}
```

Ao implementar o rastreamento do lado do cliente para métodos SGAI:

- Analise a TRACKING seção a partir das respostas da lista de ativos em vez de chamar `GetTracking`
- Use os URLs de rastreamento fornecidos na lista de ativos para relatórios de eventos publicitários
- Sinalizadores de rastreamento de disparo com base em eventos reais de reprodução de anúncios no player
- Gerencie o rastreamento de cada intervalo publicitário de forma independente à medida que as listas de ativos são buscadas

Important

A TRACKING seção só é incluída na lista de ativos quando `aws.reportingMode=CLIENT` está definida. Quando os relatórios do lado do servidor são usados (o padrão para SGAI), MediaTailor omite a TRACKING seção e, em vez disso, incorpora os dados do beacon nos URLs do anúncio. Para obter detalhes, consulte [Server-side rastreamento com inserção de anúncios guiada pelo servidor \(SGAI\)](#).

Client-side esquema e propriedades de acompanhamento de anúncios

Com o recurso de rastreamento de anúncios MediaTailor do lado do cliente, você pode integrar dados detalhados de rastreamento de anúncios do lado do cliente ao seu ambiente de player. As seções a seguir abordam o esquema geral de acompanhamento de anúncios, bem como as propriedades e valores específicos que compõem o esquema.

Schema

A estrutura JSON a seguir mostra o esquema de acompanhamento de anúncios do MediaTailor lado do cliente. Essa representação ilustra a estrutura de aninhamento do esquema para ajudá-lo a entender as relações entre as diferentes partes.

Para obter informações detalhadas sobre cada propriedade, consulte [the section called “Propriedades”](#).

```
{
  "avails": [
    {
      "ads": [
        {
          "adID": "string",
          "adParameters": "string",
          "adSystem": "string",
          "adTitle": "string",
          "adVerifications": [
            {
              "executableResource": [
                {
                  "apiFramework": "string",
                  "type": "string",
                  "uri": "string",
                  "language": "string"
                }
              ],
              "javaScriptResource": [
                {
                  "apiFramework": "string",
                  "browserOptional": "string",
                  "uri": "string"
                }
              ],
              "trackingEvents": [
```



```

        {
            "event": "string",
            "uri": "string"
        }
    ],
    "vendor": "string",
    "verificationParameters": "string"
}
],
"companionAds": [
    {
        "adParameters": "string",
        "altText": "string",
        "attributes": {
            "adSlotId": "string",
            "apiFramework": "string",
            "assetHeight": "string",
            "assetWidth": "string",
            "expandedHeight": "string",
            "expandedWidth": "string",
            "height": "string",
            "id": "string",
            "pxratio": "string",
            "renderingMode": "string",
            "width": "string"
        },
        "companionClickThrough": "string",
        "companionClickTracking": "string",
        "htmlResource": "string",
        "iFrameResource": "string",
        "sequence": "string",
        "staticResource": "string",
        "trackingEvents": [
            {
                "event": "string",
                "uri": "string"
            }
        ]
    }
],
"creativeId": "string",
"creativeSequence": "string",
"duration": "string",
"durationInSeconds": number,

```

```
"extensions": [  
  {  
    "type": "string",  
    "content": "string"  
  }  
],  
"icons": [  
  {  
    "attributes": {  
      "apiFramework": "string",  
      "duration": "string",  
      "height": "string",  
      "offset": "string",  
      "program": "string",  
      "pxratio": "string",  
      "width": "string",  
      "xPosition": "string",  
      "yPosition": "string"  
    },  
    "htmlResource": "string",  
    "iconClicks": {  
      "iconClickThrough": "string",  
      "iconClickTracking": {  
        "id": "string"  
      },  
      "iconClickFallbackImages": [  
        {  
          "altText": "string",  
          "height": "string",  
          "width": "string",  
          "staticResource": {  
            "creativeType": "string",  
            "uri": "string"  
          },  
          "uri": "string"  
        }  
      ]  
    },  
    "iconViewTracking": "string",  
    "iFrameResource": "string",  
    "staticResource": {  
      "creativeType": "string",  
      "uri": "string"  
    }  
  }  
]
```

```

],
"mediaFiles": {
  "adParameters": "string",
  "duration": "string",
  "durationInSeconds": number,
  "mediaFilesList": [
    {
      "apiFramework": "string",
      "delivery": "string",
      "height": "string",
      "maintainAspectRatio": "string",
      "mediaFileUri": "string",
      "mediaType": "string",
      "scalable": "string",
      "width": "string",
      "bitrate": "string"
    }
  ],
  "mezzanine": "string",
  "startTime": "string",
  "startTimeInSeconds": number,
  "trackingEvents": [
    {
      "beaconUrls": ["string"],
      "duration": "string",
      "durationInSeconds": number,
      "dateTime": "string",
      "eventId": "string",
      "eventType": "string",
      "startTime": "string",
      "startTimeInSeconds": number
    }
  ]
},
"startTime": "string",
"startTimeInSeconds": number,
"dateTime": "string",
"adBreakTrackingEvents": [...],
"vastAdId": "string"
}
],
"adType": "string",
"availID": "string",
"duration": "string",

```

```

    "durationInSeconds": number,
    "startTime": "string",
    "startTimeInSeconds": number,
    "dateTime": "string",
    "adMarkerDuration": "string",
    "adProgramDateTime": "string",
    "dashAvailabilityStartTime": "string",
    "hlsAnchorMediaSequenceNumber": "string"
  },
],
"nonLinearAvails": [
  {
    "nonLinearAds": [...],
    "nonLinearAdsList": [...]
  }
],
"nextToken": "string",
"meta": {}
}

```

Propriedades

A tabela a seguir lista as propriedades da API de rastreamento do lado do cliente, suas definições, tipos de valor e exemplos.

Propriedade	Definição	Tipo de valor	Exemplo
adID	- HLS - o número de sequência associado ao início do anúncio - DASH - o ID do período do anúncio Caminho: /avails/ads/adId Mapeamento VAST: Nenhum	String	10
adBreakTrackingEvents	Uma matriz que transporta eventos de rastreamento do VMAP da resposta VAST. Para obter mais	Array	[]

Propriedade	Definição	Tipo de valor	Exemplo
	informações, consulte a seção 2.3.3 da especificação VMAP 1.0. Caminho: /avails/ads/adBreakTrackingEvents		
adMarkerDuration	A duração do aproveitamento observada no marcador do anúncio no manifesto.	String	30
adParameters	Uma sequência de parâmetros de anúncio, do VAST VPAID, que MediaTailor passa para o player. Caminho: /avails/ads/adParameters Mapeamento VAST: VAST/Ad/Inline/Creatives/Creative/Linear/AdParameters	String	
adProgramDateTime	- HLS — a data, no formato ISO/IEC 8601:2004, que representa a primeira sequência de mídia do anúncio. - DASH - [Não especificado]	String	
ads	Uma matriz contendo os objetos do anúncio que compõem o item. Os anúncios são listados na ordem em que aparecem no manifesto. Caminho: /avails/ads	Array	[]

Propriedade	Definição	Tipo de valor	Exemplo
adSystem	O nome do sistema que veicula o anúncio.  Important Certifique-se de fornecer um valor. Se você não fornecer um valor, podem surgir problemas.	String	myADS
adTitle	O título do anúncio.	String	ad1
adVerifications	Contém os recursos e os metadados necessários para executar o código de medição de terceiros a fim de verificar a reprodução criativa. Para obter mais informações sobre essa propriedade, consulte a seção 3.16 da especificação VAST 4.2. MediaTailor suporta adVerifications como VAST 3 nós de extensão. Caminho: /avails/ads/adVerifications Mapeamento VAST: VAST/Ad/Inline/AdVerifications	Array	[]

Propriedade	Definição	Tipo de valor	Exemplo
<code>altText</code>	O texto alternativo para a imagem de um anúncio complementar. Este texto permite que jogadores com suporte de áudio descritivo para deficientes visuais leiam uma descrição da imagem. Caminho: <code>/avails/ads/companionAds/altText</code>	String	<pre>video sequence advertising sneakers</pre>
<code>apiFramework</code>	Defina como VPAID para informar ao jogador que esse anúncio é um anúncio VPAID. Pode aparecer em vários locais no esquema.	String	VPAID
<code>availID</code>	- HLS - o número de sequência associado ao início da disponibilidade do anúncio. - DASH - o ID do período de disponibilidade do anúncio, que geralmente é o ID do período do conteúdo que deve ser substituído por um anúncio. Caminho: <code>/avails/availID</code>	String	34 - PT34S_1
<code>avails</code>	Uma matriz contendo objetos ad-break, ou avails, que são apresentados na janela ativa do manifesto. Os itens são listados na ordem em que aparecem no manifesto. Caminho: <code>/avails</code>	Array	<code>[]</code>

Propriedade	Definição	Tipo de valor	Exemplo
adType	O tipo do anúncio. Caminho: /avails/adType e /avails/ads/adType	String	
dateTime	Programe a data e hora, no formato ISO 8601 segundos, para o início da disponibilidade do anúncio ou anúncio. Caminho: /avails/dateTime e /avails/ads/dateTime	String	
duration	Duração, no formato ISO 8601 segundos. A resposta inclui durações para toda a disponibilidade do anúncio e para cada anúncio e beacon, embora as durações dos beacons sejam sempre zero. Caminho: /avails/duration e /avails/ads/duration	String	15.015
durationInSeconds	Duração, em formato de segundos. Caminho: /avails/durationInSeconds e /avails/ads/durationInSeconds	Número	

Propriedade	Definição	Tipo de valor	Exemplo
<code>extensions</code>	Extensões personalizadas do VAST que os servidores de anúncios usam. Para obter mais informações sobre extensões, consulte a seção 3.18 da especificação VAST 4.2. Caminho: <code>/avails/ads/extensions</code> Mapeamento VAST: VAST/Ad/Inline/Extensions	Array	<code>[]</code>
<code>icons</code>	Elementos de ícone para o anúncio. Caminho: <code>/avails/ads/icons</code> Mapeamento VAST: VAST/Ad/Inline/Creatives/Creative/Linear/Icons	Array	
<code>mediaFiles</code>	Vídeo e outros recursos de que o player precisa para que o anúncio seja disponibilizado. Caminho: <code>/avails/ads/mediaFiles</code>	Objeto	
<code>nonLinearAvails</code>	Matriz de objetos não lineares e disponíveis. Caminho: <code>/nonLinearAvails</code>	Array	

Propriedade	Definição	Tipo de valor	Exemplo
<code>executableResource</code>	Recursos executáveis para verificação. Caminho: <code>/avails/ads/adVerifications/executableResource</code> Mapeamento VAST: <code>VAST/Ad/InLine/AdVerifications/Verification/ExecutableResource</code>	Array	
<code>javascriptResource</code>	JavaScript recursos para verificação. Caminho: <code>/avails/ads/adVerifications/javascriptResource</code> Mapeamento VAST: <code>VAST/Ad/InLine/AdVerifications/Verification/JavaScriptResource</code>	Array	
<code>trackingEvents</code>	Rastreamento de eventos para verificação ou anúncios complementares. Caminho: <code>/avails/ads/adVerifications/trackingEvents</code> ou <code>/avails/ads/companionAds/trackingEvents</code>	Array	

Propriedade	Definição	Tipo de valor	Exemplo
vendor	Fornecedor de verificação. Caminho: /avails/ads/adVerifications/vendor Mapeamento VAST: VAST/Ad/InLine/AdVerifications/Verification/@vendor	String	
uri	URI que aponta para um ativo executável, um ativo de vídeo ou um endpoint de rastreamento. Caminho: vários locais no esquema Mapeamento VAST: vários elementos CDATA no VAST	String	https://tracking.example.com/impression
verificationParameters	Parâmetros de verificação. Caminho: /avails/ads/adVerifications/verificationParameters Mapeamento VAST: VAST/Ad/InLine/AdVerifications/Verification/VerificationParameters	String	
attributes	Atributos de anúncios complementares, como dimensões e modo de renderização. Caminho: /avails/ads/companionAds/attributes	Objeto	

Propriedade	Definição	Tipo de valor	Exemplo
<code>companionClickThrough</code>	Um URL para a página do anunciante e que o media player abre quando o espectador clica no anúncio complementar. Caminho: <code>/avails/ads/companionAds/companionClickThrough</code> Mapeamento VAST: <code>VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/CompanionClickThrough</code>	String	<code>https://aws.amazon.com/</code>
<code>companionClickTracking</code>	O URL de rastreamento da <code>companionClickThrough</code> propriedade. Caminho: <code>/avails/ads/companionAds/companionClickTracking</code> Mapeamento VAST: <code>VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/CompanionClickTracking</code>	String	<code>https://myads.com/beaconing/event=clicktracking</code>

Propriedade	Definição	Tipo de valor	Exemplo
htmlResource	O CDATA-encoded HTML inserido diretamente na página HTML do provedor de streaming. Caminho: /avails/ads/companionAds/htmlResource Mapeamento VAST: VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/HTMLResource	String	<pre><![CDATA[<!doctype html><htm l><head>< meta name= \"viewport \" content= \"width=1, initial-s cale=1.0, minimum-s cale=1.0, ...]]></pre>
iFrameResource	O URL de um arquivo de recurso HTML que o provedor de streaming carrega em um iframe. Caminho: /avails/ads/companionAds/iFrameResource Mapeamento VAST: VAST/Ad/Inline/Creatives/Creative/CompanionAds/Companion/iFrameResource	String	
sequence	O valor da sequência especificado para o criativo na resposta VAST. Caminho: /avails/ads/companionAds/sequence	String	1

Propriedade	Definição	Tipo de valor	Exemplo
<code>startTime</code>	A posição da hora, no formato ISO 8601 segundos. Para o HLS, isso é relativo ao início da sessão de reprodução. Para o DASH, isso é relativo ao AST (Hora de Início da Disponibilidade) do manifesto. A resposta inclui horários de início de toda a disponibilidade de anúncios e de cada anúncio e beacon. Caminho: <code>/avails/startTime</code> e <code>/avails/ads/startTime</code>	String	PT18.581355S
<code>startTimeInSeconds</code>	A posição da hora, em formato de segundos. Para o HLS, isso é relativo ao início da sessão de reprodução. Para o DASH, isso é relativo ao AST (Hora de Início da Disponibilidade) do manifesto. A resposta inclui horários de início de toda a disponibilidade de anúncios e de cada anúncio e beacon. Caminho: <code>/avails/startTimeInSeconds</code> e <code>/avails/ads/startTimeInSeconds</code>	Número	18.581
<code>eventId</code>	- HLS - o número de sequência associado ao farol. - DASH - o <code>ptsTime</code> do início do anúncio.	String	

Propriedade	Definição	Tipo de valor	Exemplo
event	O nome do evento de rastreamento. Caminho: /avails/ads/adVerifications/trackingEvents/event ou /avails/ads/companionAds/trackingEvents/event	String	impression , start, firstQuartile , midpoint, thirdQuartile , complete
beaconUrls	O URL para o qual MediaTailor envia o sinalizador de anúncios. Caminho: /avails/ads/mediaFiles/trackingEvents/beaconUrls	Array	
bitrate	A taxa de bits do ativo de vídeo. Essa propriedade normalmente não é incluída em um ativo executável.	String	2048
companionAds	Uma ou mais especificações complementares de conteúdo publicitário, cada uma das quais especifica um arquivo de recurso a ser usado. Os anúncios complementares acompanham a disponibilidade do anúncio e fornecem conteúdo, como uma moldura ao redor do anúncio ou um banner, para exibição próximo ao vídeo. Caminho: /avails/ads/companionAds	Array	[]
creativeId	O valor do Id atributo da Creative tag do anúncio.	String	creative-1

Propriedade	Definição	Tipo de valor	Exemplo
<code>creativeSequence</code>	A sequência na qual um anúncio deve ser reproduzido, de acordo com o <code>AdId</code> valor na resposta VAST.	String	1
<code>dashAvailabilityStartTime</code>	Para o live/dynamic DASH, o manifesto MPD <code>availabilityStartTime</code> da origem.	String	2022-10-05T19:38:39.263Z
<code>delivery</code>	Indica se um streaming protocolo <code>progressive</code> ou está sendo usado.	String	progressive
<code>eventType</code>	O tipo de farol. Caminho: <code>/avails/ads/mediaFiles/trackingEvents/eventType</code>	String	impression
<code>height</code>	A altura, em pixels, do ativo de vídeo.	String	360
<code>hlsAnchorMediaSequenceNumber</code>	O número da sequência de mídia da sequência de <code>first/oldest</code> mídia vista no manifesto de origem do HLS.	String	77
<code>maintainAspectRatio</code>	Indica se a proporção do vídeo deve ser mantida durante o dimensionamento.	Booleano	true
<code>mediaFilesList</code>	Especifica o vídeo e outros recursos de que o player precisa para que o anúncio seja disponibilizado. Caminho: <code>/avails/ads/mediaFiles/mediaFilesList</code>	Array	[]

Propriedade	Definição	Tipo de valor	Exemplo
<code>mediaFileUri</code>	URI que aponta para um ativo executável ou um ativo de vídeo.	String	<code>https://myad.com/ad/ad134/vpaid.js</code>
<code>mediaType</code>	O tipo MIME do ativo criativo ou complementar.	String	<code>video/mp4</code>
<code>meta</code>	Metadados adicionais para o anúncio.	Objeto	
<code>mezzanine</code>	O URL do ativo MP4 mezanino, especificado se o anúncio VPAID incluir um. Caminho: <code>/avails/ads/mediaFiles/mezzanine</code>	String	<code>https://gcdn.2mdn.net/video/playback/id/itag/c/k2/file/file.mp4</code>
<code>nextToken</code>	O valor do token que aponta para a próxima página de resultados, quando esse valor existe.	String	<code>UFQz0S44NzNTXzIwMjMtMDctMzFUMTY6NTA6MDYuMzUwNjI2ODQ1Wl8x</code>
<code>nonLinearAds</code>	Non-linear anúncios que aparecem ao lado do conteúdo do vídeo.	Array	<code>[]</code>
<code>nonLinearAdsList</code>	Lista de anúncios não lineares.	Array	<code>[]</code>
<code>scalable</code>	Indica se o vídeo deve ser dimensionado para outras dimensões.	Booleano	<code>true</code>

Propriedade	Definição	Tipo de valor	Exemplo
<code>skipOffset</code>	O valor do tempo que identifica quando o jogador disponibiliza os controles de salto para o usuário.	String	<code>00:00:05</code>
<code>staticResource</code>	O URL de um arquivo criativo estático usado para o componente de anúncio. Caminho: <code>/avails/ads/companionAds/staticResource</code>	String	<code>https://very-interactive-ads.com/campaign1/file.json?c=1019113602</code>
<code>vastAdId</code>	O valor do Id atributo da Ad tag.	String	<code>ad1</code>
<code>width</code>	A largura, em pixels, do ativo de vídeo.	String	<code>640</code>
<code>xPosition</code>	A posição horizontal de um ícone dentro do player de vídeo. Pode ser um valor de pixel específico ou uma posição como “esquerda” ou “direita”. Caminho: <code>/avails/ads/icons/attributes/xPosition</code>	String	<code>left</code> ou <code>10</code>
<code>yPosition</code>	A posição vertical de um ícone no player de vídeo. Pode ser um valor de pixel específico ou uma posição como “superior” ou “inferior”. Caminho: <code>/avails/ads/icons/attributes/yPosition</code>	String	<code>top</code> ou <code>10</code>

Propriedade	Definição	Tipo de valor	Exemplo
<code>iconClicks</code>	Contém informações de clique e rastreamento de um ícone. Caminho: <code>/avails/ads/icons/iconClicks</code>	Objeto	
<code>iconClickThrough</code>	Um URL para a página do anunciante e que o media player abre quando o espectador clica no ícone. Caminho: <code>/avails/ads/icons/iconClicks/iconClickThrough</code>	String	<code>https://advertiser.com/landing-page</code>
<code>iconClickTracking</code>	O URL de rastreamento da <code>iconClickThrough</code> propriedade. Caminho: <code>/avails/ads/icons/iconClicks/iconClickTracking</code>	Objeto	
<code>iconClickFallbackImages</code>	Uma matriz de imagens alternativas a serem exibidas se o ícone não puder ser exibido. Caminho: <code>/avails/ads/icons/iconClicks/iconClickFallbackImages</code>	Array	
<code>iconViewTracking</code>	O URL para rastrear quando um ícone é visualizado. Caminho: <code>/avails/ads/icons/iconViewTracking</code>	String	<code>https://tracking.example.com/icon-view</code>

Propriedade	Definição	Tipo de valor	Exemplo
offset	O intervalo de tempo em que um ícone deve aparecer durante a reprodução do anúncio. Caminho: /avails/ads/icons/attributes/offset	String	00:00:05
program	O programa ou iniciativa associado ao ícone, como "AdChoices". Caminho: /avails/ads/icons/attributes/program	String	AdChoices
pxratio	A proporção de pixels do ícone ou anúncio complementar, usada para exibições de alto DPI. Caminho: /avails/ads/icons/attributes/pxratio ou /avails/ads/companionAds/attributes/pxratio	String	1 ou 2
type	O tipo de recurso ou extensão. Caminho: /avails/ads/extensions/type ou /avails/ads/adVerifications/executableResource/type	String	text/javascript
content	O conteúdo de uma extensão. Caminho: /avails/ads/extensions/content	String	

Propriedade	Definição	Tipo de valor	Exemplo
language	A linguagem de programação de um recurso executável. Caminho: /avails/ads/adVerifications/executableResource/language	String	javascript
browserOptional	Indica se o suporte do navegador é necessário para o JavaScript recurso. Caminho: /avails/ads/adVerifications/javascriptResource/browserOptional	String	true ou false
id	Um identificador para vários elementos no esquema. Caminho: /avails/ads/companionAds/attributes/id ou /avails/ads/icons/iconClicks/iconClickTracking/id	String	companion-1
assetHeight	A altura do ativo publicitário complementar. Caminho: /avails/ads/companionAds/attributes/assetHeight	String	250

Propriedade	Definição	Tipo de valor	Exemplo
<code>assetWidth</code>	A largura do ativo do anúncio complementar. Caminho: <code>/avails/ads/companionAds/attributes/assetWidth</code>	String	<code>300</code>
<code>expandedHeight</code>	A altura do anúncio complementar quando expandido. Caminho: <code>/avails/ads/companionAds/attributes/expandedHeight</code>	String	<code>600</code>
<code>expandedWidth</code>	A largura do anúncio complementar quando expandido. Caminho: <code>/avails/ads/companionAds/attributes/expandedWidth</code>	String	<code>600</code>
<code>renderingMode</code>	O modo de renderização do anúncio complementar. Caminho: <code>/avails/ads/companionAds/attributes/renderingMode</code>	String	<code>default</code> ou <code>transparent</code>
<code>adSlotId</code>	O ID do espaço publicitário em que o anúncio complementar deve ser exibido. Caminho: <code>/avails/ads/companionAds/attributes/adSlotId</code>	String	<code>banner-1</code>

Propriedade	Definição	Tipo de valor	Exemplo
<code>creativeType</code>	O tipo MIME do ativo criativo. Caminho: <code>/avails/ads/icons/staticResource/creativeType</code>	String	<code>image/png</code>

Ad-tracking tempo de atividade

Com os relatórios do lado do cliente, o jogador deve emitir eventos de rastreamento (beacons) com um nível de precisão. Usando o esquema de rastreamento MediaTailor do lado do cliente, você pode garantir que, para cada disponibilidade, anúncio, complemento, sobreposição e eventos de rastreamento, as informações de tempo e duração estejam presentes e em formas diferentes.

Use os seguintes MediaTailor key/value pares para que o player concilie com precisão as atividades de eventos publicitários, como eventos de rastreamento, com a posição de reprodução:

- [`startTime`](#)
- [`startTimeInSeconds`](#)
- [`adProgramDateTime`](#)
- [`adID/eventId`](#)

O HLS e o DASH implementam o valor de `startTime` e `startTimeInSeconds` de forma diferente:

- HLS - Os `startTime` valores são relativos ao início da sessão de reprodução. O início da sessão de reprodução é definido como hora zero. O anúncio `startTime` é a soma dos valores cumulativos de todas as durações do EXT-INF segmento até a disponibilidade. O número de sequência de mídia do segmento em que o anúncio ou o evento de rastreamento se enquadra também corresponde à `adId` ou `eventId` na resposta de rastreamento do lado do cliente.
- DASH:
 - Live/dynamic manifestos - Os `startTime` valores são relativos aos `MPD@availabilityStartTime` do manifesto DASH. `MPD@availabilityStartTime` é uma âncora de temporização para todas as MediaTailor sessões que consomem o stream.
 - VOD/static manifestos - Os `startTime` valores são relativos ao início da sessão de reprodução. O início da sessão de reprodução é definido como hora zero. Cada anúncio dentro do avail

está contido em seu próprio `Period` elemento. O `Period` elemento tem um `@start` atributo com um valor igual aos `startTime` valores na carga de rastreamento do lado do cliente. `PeriodId` Também corresponde à `adId` ou `eventId` na resposta de rastreamento do lado do cliente.

Example HLS:

No exemplo a seguir, a MediaTailor sessão foi iniciada e o manifesto a seguir é o primeiro exibido ao cliente:

```
#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
```


Na carga útil JSON de rastreamento do lado do cliente, os seguintes valores se aplicam:

- `startTime`: "PT18.581355S"
- `startTimeInSeconds`: 18.581
- `availProgramDateTime`: "2023-05-03T21:24:41.453Z"
- `adId`: 4603269

Example DASH:

No exemplo a seguir, a MediaTailor sessão recebe uma lista intermediária no manifesto. Observe que o valor do `@start` atributo do segundo período, que é o período do anúncio, tem um valor relativo ao `MPD@availabilityStartTime` valor. Esse valor é MediaTailor gravado nos `startTime` campos de resposta de rastreamento do lado do cliente para todas as sessões.

```
<?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>
```

Na carga útil JSON de rastreamento do lado do cliente, os seguintes valores se aplicam:

- `startTime`: "PT5042H27M59.931S"
- `startTimeInSeconds`: 18152879.931
- `availProgramDateTime`: *null*
- `adId`: 1683151599194_1_1

Controles e funcionalidades do player para rastreamento de anúncios do lado do cliente

MediaTailor os metadados de rastreamento do lado do cliente oferecem suporte a vários controles e funcionalidades do player. A lista a seguir descreve os controles populares do jogador.

Tópicos

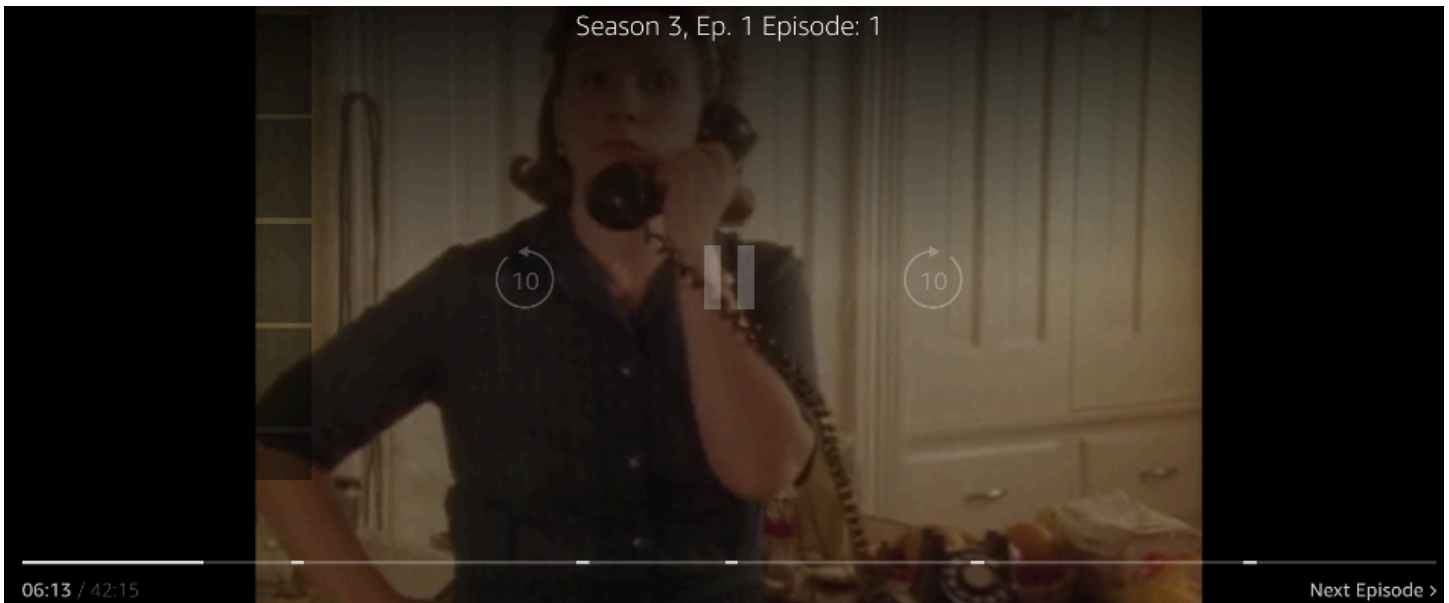
- [Esfregando](#)
- [Adicionar cronômetro de contagem regressiva](#)
- [Anúncios ignoráveis](#)
- [Clique no anúncio](#)
- [Anúncios complementares](#)
- [Anúncios interativos \(SIMID\)](#)
- [Anúncios interativos \(VPAID\)](#)
- [Ícones para o Google Por que esse anúncio \(WTA\)](#)

Esfregando

Para aprimorar a experiência de reprodução, o player pode exibir as posições do anúncio na linha do tempo de reprodução. MediaTailor disponibiliza essas posições do anúncio na forma de `startTimeInSeconds` valores na resposta de rastreamento do lado do cliente.

Note

Alguns provedores de streaming evitam ultrapassar a posição de um anúncio.

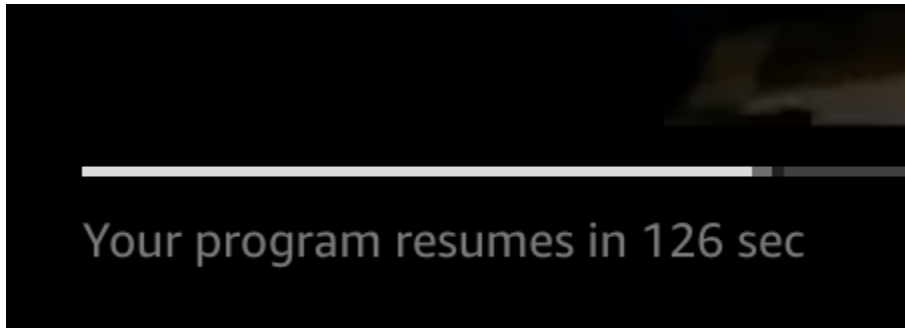


A seguinte resposta JSON da carga útil de rastreamento do lado do cliente mostra a hora de início do avial (ad break) dentro do objeto JSON raiz da matriz avails. O player usa esses dados para mostrar a localização do intervalo publicitário na linha do tempo do jogador, em 28 segundos.

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

Adicionar cronômetro de contagem regressiva

Com MediaTailor você pode usar um cronômetro de contagem regressiva de anúncios para ajudar a manter seu público envolvido durante a visualização do intervalo de anúncios. O público pode usar o cronômetro para entender quando o intervalo publicitário termina e o programa é retomado.



Os elementos nos metadados de rastreamento do lado do cliente que desempenham um papel no cronômetro de contagem regressiva do anúncio são `startTime`, `e. startTimeInSeconds` `duration` `durationInSeconds`. O player usa esses metadados, junto com o tempo decorrido da sessão que ele rastreia separadamente, para determinar quando exibir o cronômetro e o valor a partir do qual ele deve ser contado.

A seguinte resposta JSON da carga útil de rastreamento do lado do cliente mostra as informações necessárias para exibir um cronômetro de contagem regressiva de anúncios.

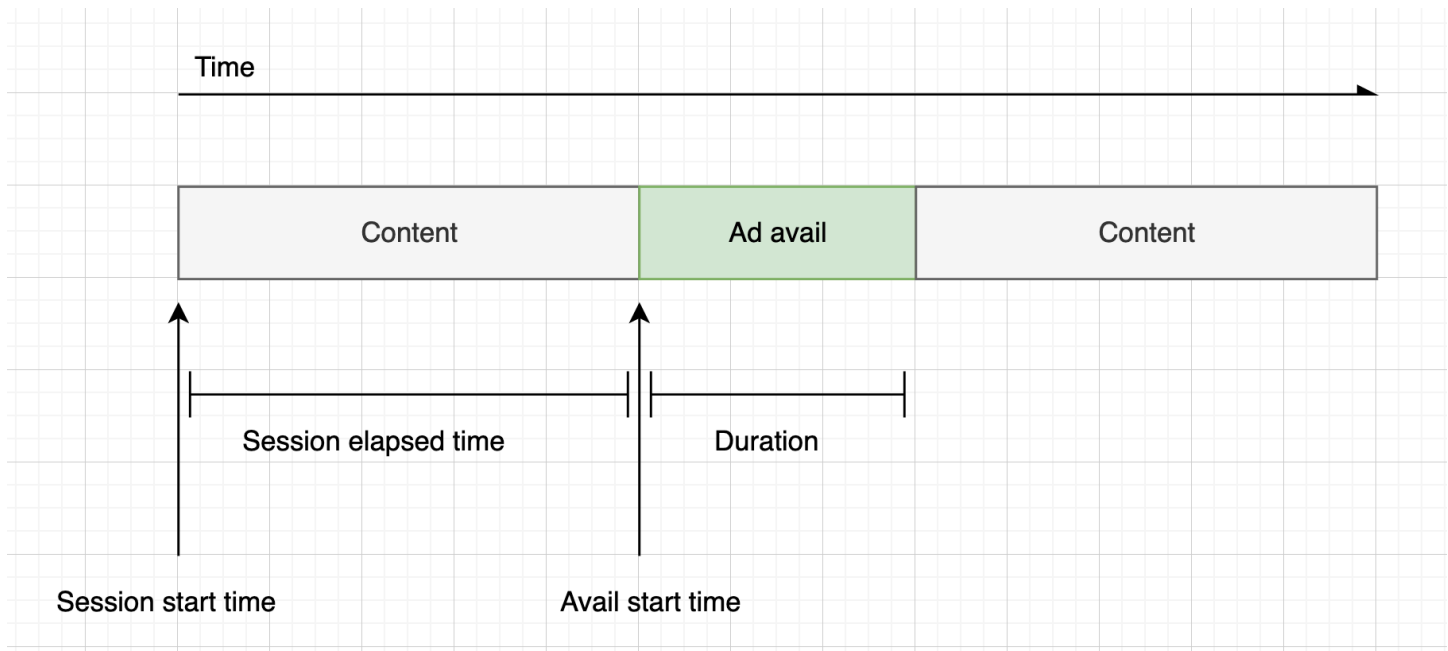
```
{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [...],
      "availId": "7",
      "availProgramDateTime": null,
      "duration": "PT30S",
      "durationInSeconds": 30,
      "meta": null,
      "nonLinearAdsList": [],
      "startTime": "PT28S",
      "startTimeInSeconds": 28
    }
  ],
  "dashAvailabilityStartTime": null,
  "hlsAnchorMediaSequenceNumber": null,
  "nextToken": "UFQxMk0zNC44NjhTXzIwMjMtMDctMjFUMjA6MjM6MDcuNzc1NzE2MzAyWl8x",
}
```

```
"nonLinearAvails": []
}
```

Quando o tempo decorrido da sessão atinge o horário de início do avail, o jogador exibe uma contagem regressiva com um valor que corresponde à duração do avail. O valor do cronômetro de contagem regressiva diminui à medida que o tempo decorrido avança além do horário de início do item.

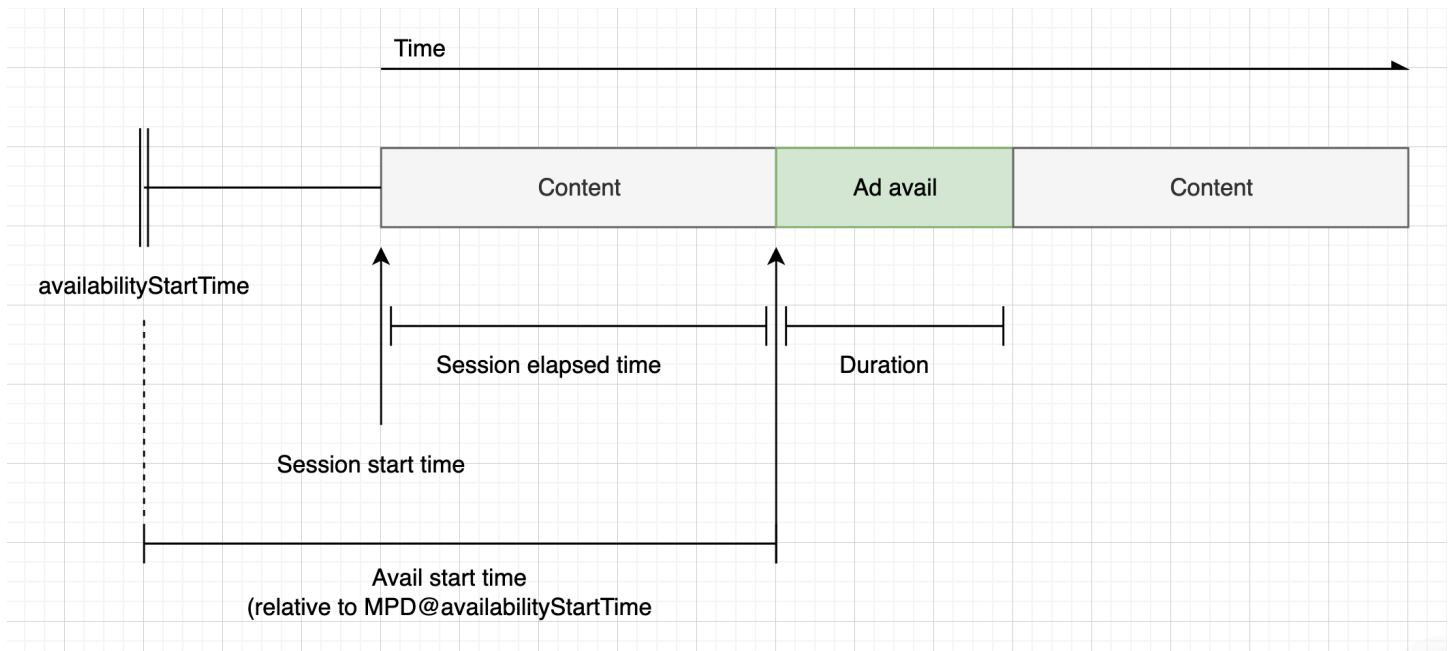
Example fórmula: cronômetro de contagem regressiva para HLS (ao vivo e VOD) e DASH (VOD)

- $\text{session\_start\_time} = \text{a soma de todos os valores de EXT-INF duração} - \text{o valor da duração das três sequências de EXT-INF mídia mais recentes}$
- $\text{valor do cronômetro} = \text{duration} - (\text{session\_elapsed\_time} - \text{startTime})$



Example fórmula: cronômetro de contagem regressiva para DASH (ao vivo)

- $\text{session\_start\_time} = (\text{segmento mais recente startTime} + \text{duration}) / \text{timescale} - \text{MPD@suggestedPresentationDelay}$
- $\text{valor do cronômetro} = \text{duration} - (\text{session\_elapsed\_time} - \text{startTime})$



Anúncios ignoráveis

Anúncios ignoráveis são espaços publicitários que permitem ao espectador pular parte do anúncio para retomar a visualização do programa. No VAST, o `Linear@skipOffset` atributo identifica um anúncio que pode ser ignorado.

A resposta VAST a seguir mostra como usar um anúncio que pode ser ignorado:

```
<?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>
```

A seguinte resposta JSON da carga útil de rastreamento do lado do cliente mostra os metadados do anúncio dentro da matriz. ads A matriz contém o skipOffset valor MediaTailor obtido da resposta VAST.

```
{
  "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": []
}

```

Clique no anúncio

Click-through Os URIs permitem que os anunciantes avaliem o sucesso de um anúncio em atrair a atenção dos espectadores. Depois que um espectador clica no quadro de vídeo ativo de um anúncio em andamento, um navegador da Web abre o URI da página inicial ou da página de destino da campanha do anunciante. O desenvolvedor do player determina o comportamento do clique, como sobrepor um botão ou rótulo no vídeo do anúncio, com uma mensagem para clicar para saber mais. Os desenvolvedores do player geralmente pausam o vídeo do anúncio depois que os espectadores clicam no quadro ativo do vídeo.



Click here for deals on Amazon.com

MediaTailor pode analisar e disponibilizar qualquer URL de evento de clique linear de vídeo retornado na resposta VAST. A resposta VAST a seguir mostra um exemplo de clique em um anúncio.

```
<?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>

```

A seguinte resposta JSON da carga útil de rastreamento do lado do cliente mostra como MediaTailor exibe os URLs de clique e rastreamento de cliques dentro da matriz. `trackingEvents` O tipo de `clickThrough` evento representa o anúncio clicável e o tipo de `clickTracking` evento representa o URL de rastreamento de cliques.

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "1",
          "adParameters": "",
          "adProgramDateTime": "2023-07-31T16:53:40.577Z",
          "adSystem": "2.0",
          "adTitle": "1",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "000006",
          "creativeSequence": "1",
          "duration": "PT14.982S",
          "durationInSeconds": 14.982,
          "extensions": [],
          "mediaFiles": {
            "mediaFilesList": [],

```

```
    "mezzanine": ""
  },
  "skipOffset": null,
  "startTime": "PT39.339S",
  "startTimeInSeconds": 39.339,
  "trackingEvents": [
    {
      "beaconUrls": [
        "https://myads.com/beaconing/event=impression"
      ],
      "duration": "PT14.982S",
      "durationInSeconds": 14.982,
      "eventId": "2698188",
      "eventProgramDateTime": null,
      "eventType": "impression",
      "startTime": "PT39.339S",
      "startTimeInSeconds": 39.339
    },
    {
      "beaconUrls": [
        "https://aws.amazon.com"
      ],
      "duration": "PT14.982S",
      "durationInSeconds": 14.982,
      "eventId": "2698188",
      "eventProgramDateTime": null,
      "eventType": "clickThrough",
      "startTime": "PT39.339S",
      "startTimeInSeconds": 39.339
    },
    {
      "beaconUrls": [
        "https://myads.com/beaconing/event=clicktracking"
      ],
      "duration": "PT14.982S",
      "durationInSeconds": 14.982,
      "eventId": "2698795",
      "eventProgramDateTime": null,
      "eventType": "clickTracking",
      "startTime": "PT39.339S",
      "startTimeInSeconds": 39.339
    }
  ],
  "vastAdId": ""
```

```

    }
  ],
  "availId": "2698188",
  "availProgramDateTime": "2023-07-31T16:53:40.577Z",
  "duration": "PT14.982S",
  "durationInSeconds": 14.982,
  "meta": null,
  "nonLinearAdsList": [],
  "startTime": "PT39.339S",
  "startTimeInSeconds": 39.339
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQz0S4zMz1TXzIwMjMtMDctMzFUMTY6NTQ6MDQu0DA1Mzk2NTI5Wl8x",
"nonLinearAvails": []
}

```

Anúncios complementares

Um anúncio complementar aparece ao lado de um criativo linear. Use anúncios complementares para aumentar a eficácia de um anúncio exibindo informações sobre produtos, logotipos e marcas. O anúncio gráfico pode apresentar códigos de resposta rápida (QR) e áreas clicáveis para promover o engajamento do público.

MediaTailor suporta anúncios complementares na resposta VAST. Ele pode passar por metadados de `StaticResource` e `FrameResource`, e `HTMLResource` nós, respectivamente.

A resposta VAST a seguir mostra um exemplo de localização e formato do anúncio linear e do anúncio complementar.

```

<?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>

```

Os dados aparecem na resposta de rastreamento do lado do cliente na `/avail/x/ads/y/companionAds` lista. Cada criação linear pode conter até 6 anúncios complementares. Conforme mostrado no exemplo abaixo, os anúncios complementares aparecem em uma lista

Note

Como prática recomendada, os desenvolvedores de aplicativos devem implementar a lógica para remover ou descarregar explicitamente o anúncio complementar no final do criativo.

```
{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "0",
          "adParameters": "",
          "adProgramDateTime": null,
          "adSystem": "EMT",
          "adTitle": "sample",
          "adVerifications": [],
          "companionAds": [
            {
              "adParameters": null,
              "altText": null,
              "attributes": {
                "adSlotId": null,
                "apiFramework": null,
                "assetHeight": null,
                "assetWidth": null,
                "expandedHeight": null,
                "expandedWidth": null,
                "height": "250",
                "id": "2",
                "pxratio": null,
                "renderingMode": null,
                "width": "300"
              },
              "companionClickThrough": "https://beacon.com/2",
              "companionClickTracking": null,
              "htmlResource": null,
              "iFrameResource": null,
              "sequence": "1",
              "staticResource": "https://emt.com/companion/9973499273",
            }
          ],
          "sequence": "1",
          "staticResource": "https://emt.com/companion/9973499273",
        }
      ],
      "sequence": "1",
      "staticResource": "https://emt.com/companion/9973499273",
    }
  ],
  "sequence": "1",
  "staticResource": "https://emt.com/companion/9973499273",
}
```

```
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://beacon.com/1"
        ],
        "eventType": "creativeView"
      }
    ],
  },
  {
    "adParameters": null,
    "altText": null,
    "attributes": {
      "adSlotId": null,
      "apiFramework": null,
      "assetHeight": null,
      "assetWidth": null,
      "expandedHeight": null,
      "expandedWidth": null,
      "height": "90",
      "id": "3",
      "pxratio": null,
      "renderingMode": null,
      "width": "728"
    },
    "companionClickThrough": "https://beacon.com/4",
    "companionClickTracking": null,
    "htmlResource": null,
    "iFrameResource": null,
    "sequence": "1",
    "staticResource": "https://emt.com/companion/1238901823",
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://beacon.com/3"
        ],
        "eventType": "creativeView"
      }
    ]
  }
],
"creativeId": "1",
"creativeSequence": "1",
"duration": "PT10S",
```

```
    "durationInSeconds": 10,
    "extensions": [],
    "mediaFiles": {
      "mediaFilesList": [],
      "mezzanine": ""
    },
    "skipOffset": null,
    "startTime": "PT0S",
    "startTimeInSeconds": 0,
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://beacon.com/impression/1"
        ],
        "duration": "PT10S",
        "durationInSeconds": 10,
        "eventId": "0",
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT0S",
        "startTimeInSeconds": 0
      }
    ],
    "vastAdId": ""
  }
],
"availId": "0",
"availProgramDateTime": null,
"duration": "PT10S",
"durationInSeconds": 10,
"meta": null,
"nonLinearAdsList": [],
"startTime": "PT0S",
"startTimeInSeconds": 0
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQxMFNFmJyAyMy0wNy0wNlQyMToxMDowOC42NzQ4NDA1NjJaXzE%3D",
"nonLinearAvails": []
}
```


Anúncios interativos (SIMID)

SecureInteractive O Media Interface Definition (SIMID) é um padrão para publicidade interativa que foi introduzido no padrão VAST 4.x do Interactive Advertising Bureau (IAB). O SIMID separa o carregamento de elementos interativos do criativo linear primário no player, referenciando ambos na resposta VAST. MediaTailor insere o criativo principal para manter a experiência de reprodução e coloca os metadados dos componentes interativos na resposta de rastreamento do lado do cliente.

No exemplo de resposta VAST 4 a seguir, a carga útil do SIMID está dentro do InteractiveCreativeFile nó.

```
<?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>
```

```

    </Linear>
  </Creative>
</Creatives>
</InLine>
</Ad>
</VAST>

```

Na resposta VAST 3 a seguir, a carga útil do SIMID está dentro do Extensions nó.

```

<?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>

```

Na seguinte resposta de rastreamento do lado do cliente, os dados do SIMID aparecem na lista. /
avails/x/ads/y/extensions

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "1",
          "adParameters": "",
          "adProgramDateTime": "2023-07-31T16:53:40.577Z",
          "adSystem": "2.0",
          "adTitle": "Linear SIMID Example",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "1",
          "creativeSequence": "1",
          "duration": "PT14.982S",
          "durationInSeconds": 14.982,
          "extensions": [
            {
              "content": "<InteractiveCreativeFile type=\"text/html\" apiFramework=\\\"SIMID\\\" variableDuration=\\\"true\\\">\\nhttps://interactive-ads.com/interactive-media-ad-sample/sample_simid.html</InteractiveCreativeFile>",
              "type": "InteractiveCreativeFile"
            }
          ],
          "mediaFiles": {
            "mediaFilesList": [],
            "mezzanine": ""
          },
          "skipOffset": null,
          "startTime": "PT39.339S",

```

```
"startTimeInSeconds": 39.339,
"trackingEvents": [
  {
    "beaconUrls": [
      "https://myads.com/beaconing/event=impression"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698188",
    "eventProgramDateTime": null,
    "eventType": "impression",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  },
  {
    "beaconUrls": [
      "https://aws.amazon.com"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698188",
    "eventProgramDateTime": null,
    "eventType": "clickThrough",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  },
  {
    "beaconUrls": [
      "https://myads.com/beaconing/event=clicktracking"
    ],
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "eventId": "2698795",
    "eventProgramDateTime": null,
    "eventType": "clickTracking",
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  }
],
"vastAdId": ""
}
],
"availId": "2698188",
"availProgramDateTime": "2023-07-31T16:53:40.577Z",
```

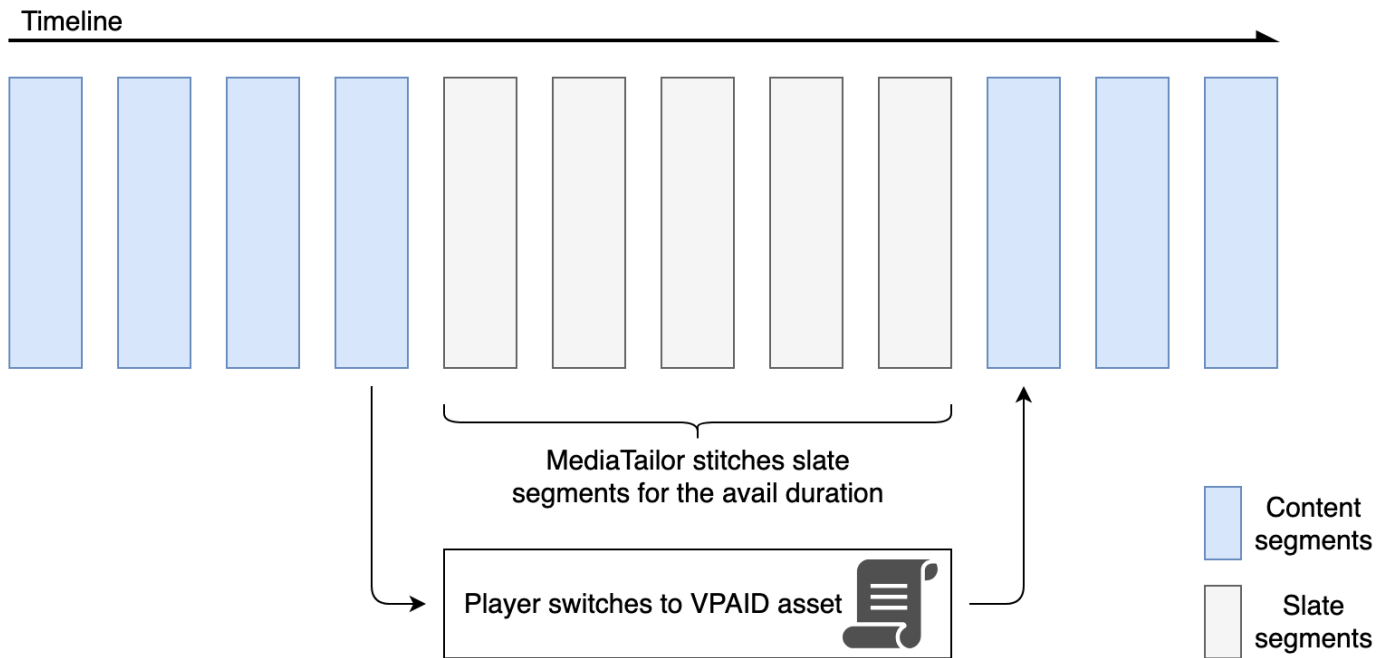
```
    "duration": "PT14.982S",
    "durationInSeconds": 14.982,
    "meta": null,
    "nonLinearAdsList": [],
    "startTime": "PT39.339S",
    "startTimeInSeconds": 39.339
  }
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQz0S4zMz1TXzIwMjMtMDctMzFUMTY6NTQ6MDQu0DA1Mzk2NTI5Wl8x",
"nonLinearAvails": []
}
```

Anúncios interativos (VPAID)

A definição de interface de anúncio do player de vídeo (VPAID) especifica o protocolo entre o anúncio e o player de vídeo que permite a interatividade do anúncio e outras funcionalidades. Para transmissões ao vivo, MediaTailor suporta o formato VPAID agrupando segmentos de ardósia durante a disponibilidade e colocando metadados para os criativos do VPAID na resposta de rastreamento do lado do cliente que o player de vídeo consome. O player baixa os arquivos VPAID, reproduz o criativo linear e executa os scripts do cliente. O jogador nunca deve jogar os segmentos de ardósia.

Note

O VPAID está obsoleto a partir do VAST 4.1.



O exemplo a seguir mostra o conteúdo VPAID na resposta VAST.

```
<?xml version="1.0"?>
<VAST xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="vast.xsd" version="3.0">
  <Ad id="1234567">
    <Inline>
      <AdSystem>GDFP</AdSystem>
      <AdTitle>VPAID</AdTitle>
      <Description>Vpaid Linear Video Ad</Description>
      <Error>http://www.example.com/error</Error>
      <Impression>http://www.example.com/impression</Impression>
      <Creatives>
        <Creative sequence="1">
          <Linear>
            <Duration>00:00:00</Duration>
            <TrackingEvents>
              <Tracking event="start">http://www.example.com/start</Tracking>
              <Tracking event="firstQuartile">http://www.example.com/firstQuartile</
Tracking>
              <Tracking event="midpoint">http://www.example.com/midpoint</Tracking>
              <Tracking event="thirdQuartile">http://www.example.com/thirdQuartile</
Tracking>
```

```

    <Tracking event="complete">http://www.example.com/complete</Tracking>
    <Tracking event="mute">http://www.example.com/mute</Tracking>
    <Tracking event="unmute">http://www.example.com/unmute</Tracking>
    <Tracking event="rewind">http://www.example.com/rewind</Tracking>
    <Tracking event="pause">http://www.example.com/pause</Tracking>
    <Tracking event="resume">http://www.example.com/resume</Tracking>
    <Tracking event="fullscreen">http://www.example.com/fullscreen</Tracking>
    <Tracking event="creativeView">http://www.example.com/creativeView</
Tracking>
    <Tracking event="acceptInvitation">http://www.example.com/
acceptInvitation</Tracking>
  </TrackingEvents>
  <AdParameters><![CDATA[ {"videos":[ {"url":"https://my-ads.com/interactive-
media-ads/media/media_linear_VPAID.mp4","mimetype":"video/mp4"}]] ]]></AdParameters>
  <VideoClicks>
    <ClickThrough id="123">http://google.com</ClickThrough>
    <ClickTracking id="123">http://www.example.com/click</ClickTracking>
  </VideoClicks>
  <MediaFiles>
    <MediaFile delivery="progressive" apiFramework="VPAID" type="application/
javascript" width="640" height="480"> https://googleads.github.io/googleads-ima-html5/
vpaid/linear/VpaidVideoAd.js </MediaFile>
  </MediaFiles>
</Linear>
</Creative>
</Creatives>
</InLine>
</Ad>
</VAST>

```

O exemplo a seguir mostra as informações de rastreamento.

```

{
  "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": "UFQz0S4zMzlTXzIwMjMtMDctMzFUMTY6NTQ6MDQu0DA1Mzk2NTI5Wl8x",
    "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": "UFQ4TTQyLjI4OVNfMjAyMy0wOC0xNFQxOT01MD00MS4zOTc5MjAzODVaXzE%3D",
"nonLinearAvails": []
}

```

Ícones para o Google Por que esse anúncio (WTA)

AdChoices é um padrão do setor que fornece aos espectadores informações sobre os anúncios que eles veem, incluindo como esses anúncios foram segmentados para eles.



A API de rastreamento MediaTailor do lado do cliente suporta metadados de ícones transportados no nó de extensões VAST da resposta VAST. Para obter mais informações sobre o WTA na resposta VAST, consulte [este exemplo de resposta VAST XML](#).

Note

MediaTailor atualmente suporta apenas a versão 3 do VAST.

```

<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>

```

O exemplo a seguir mostra a resposta de rastreamento do lado do cliente na `/avails/x/ads/y/extensions` lista.

```

{
  "avails": [
    {
      "adBreakTrackingEvents": [],
      "adMarkerDuration": null,
      "ads": [
        {
          "adId": "0",
          "adParameters": "",
          "adProgramDateTime": null,
          "adSystem": "GDFP",
          "adTitle": "Google Why This Ad VAST 3 Sample",
          "adVerifications": [],

```

```

    "companionAds": [],
    "creativeId": "7891011",
    "creativeSequence": "1",
    "duration": "PT10S",
    "durationInSeconds": 10,
    "extensions": [
      {
        "content": "<IconClickFallbackImages program=\"GoogleWhyThisAd\">
          <IconClickFallbackImage height=\"150\" width=\"400\">
            <AltText>Alt icon fallback</AltText>
            <StaticResource creativeType=\"image/png\"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png]]>
          </StaticResource>
        </IconClickFallbackImage>
      </IconClickFallbackImages>
      <IconClickFallbackImages program=\"AdChoices\">
        <IconClickFallbackImage height=\"150\" width=\"400\">
          <AltText>Alt icon fallback</AltText>
          <StaticResource creativeType=\"image/png\"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png?size=1x]]>
        </StaticResource>
      </IconClickFallbackImage>
        <IconClickFallbackImage height=\"300\" width=\"800\">
          <AltText>Alt icon fallback</AltText>
          <StaticResource creativeType=\"image/png\"><![CDATA[https://
storage.googleapis.com/interactive-media-ads/images/wta_dialog.png?size=2x]]>
        </StaticResource>
      </IconClickFallbackImage>
    </IconClickFallbackImages>",
        "type": "IconClickFallbackImages"
      }
    ],
    "mediaFiles": {
      "mediaFilesList": [],
      "mezzanine": ""
    },
    "skipOffset": "00:00:03",
    "startTime": "PT0S",
    "startTimeInSeconds": 0,
    "trackingEvents": [
      {
        "beaconUrls": [
          "https://example.com/view"
        ]
      }
    ],

```

```

        "duration": "PT10S",
        "durationInSeconds": 10,
        "eventId": "0",
        "eventProgramDateTime": null,
        "eventType": "impression",
        "startTime": "PT0S",
        "startTimeInSeconds": 0
      }
    ],
    "vastAdId": "123456"
  }
],
"availId": "0",
"availProgramDateTime": null,
"duration": "PT10S",
"durationInSeconds": 10,
"meta": null,
"nonLinearAdsList": [],
"startTime": "PT0S",
"startTimeInSeconds": 0
}
],
"dashAvailabilityStartTime": null,
"hlsAnchorMediaSequenceNumber": null,
"nextToken": "UFQxMFNfMjAyMy0wNy0wNlQyMDo0MT0xNy45NDE4MDM0NDhaXzE%3D",
"nonLinearAvails": []
}

```

Client-side balizamento

Com o `startTimeInSeconds` elemento de rastreamento do lado do cliente, você pode usá-lo MediaTailor para apoiar a temporização dos beacons.

A resposta JSON a seguir mostra os principais tipos de beacon: impressões, início, quartis e conclusão.

Note

As diretrizes de medição de impressão em vídeo do Interactive Advertising Bureau (IAB) afirmam que uma impressão exige que o conteúdo do anúncio seja carregado no lado do cliente e, no mínimo, comece a ser renderizado no player. Para obter mais informações, consulte [Modelo de veiculação de anúncios em vídeo digital \(VAST\)](#) no site do IAB.

```
{
  "avails": [
    {
      "ads": [
        {
          "adId": "8104385",
          "duration": "PT15.100000078S",
          "durationInSeconds": 15.1,
          "startTime": "PT17.817798612S",
          "startTimeInSeconds": 17.817,
          "trackingEvents": [
            {
              "beaconUrls": [
                "http://exampleleadserver.com/tracking?event=impression"
              ],
              "duration": "PT15.100000078S",
              "durationInSeconds": 15.1,
              "eventId": "8104385",
              "eventType": "impression",
              "startTime": "PT17.817798612S",
              "startTimeInSeconds": 17.817
            },
            {
              "beaconUrls": [
                "http://exampleleadserver.com/tracking?event=start"
              ],
              "duration": "PT0S",
              "durationInSeconds": 0.0,
              "eventId": "8104385",
              "eventType": "start",
              "startTime": "PT17.817798612S",
              "startTimeInSeconds": 17.817
            },
            {
              "beaconUrls": [
                "http://exampleleadserver.com/tracking?event=firstQuartile"
              ],
              "duration": "PT0S",
              "durationInSeconds": 0.0,
              "eventId": "8104386",
              "eventType": "firstQuartile",
              "startTime": "PT21.592798631S",
              "startTimeInSeconds": 21.592
            }
          ]
        }
      ]
    }
  ]
}
```

```
    },
    {
      "beaconUrls": [
        "http://exampleleadserver.com/tracking?event=midpoint"
      ],
      "duration": "PT0S",
      "durationInSeconds": 0.0,
      "eventId": "8104387",
      "eventType": "midpoint",
      "startTime": "PT25.367798651S",
      "startTimeInSeconds": 25.367
    },
    {
      "beaconUrls": [
        "http://exampleleadserver.com/tracking?event=thirdQuartile"
      ],
      "duration": "PT0S",
      "durationInSeconds": 0.0,
      "eventId": "8104388",
      "eventType": "thirdQuartile",
      "startTime": "PT29.14279867S",
      "startTimeInSeconds": 29.142
    },
    {
      "beaconUrls": [
        "http://exampleleadserver.com/tracking?event=complete"
      ],
      "duration": "PT0S",
      "durationInSeconds": 0.0,
      "eventId": "8104390",
      "eventType": "complete",
      "startTime": "PT32.91779869S",
      "startTimeInSeconds": 32.917
    }
  ]
}
],
"availId": "8104385",
"duration": "PT15.100000078S",
"durationInSeconds": 15.1,
"startTime": "PT17.817798612S",
"startTimeInSeconds": 17.817
}
```



```
}
```

Modo híbrido com beacons de anúncios do lado do servidor

MediaTailor suporta um modo híbrido para rastreamento de sessões. Nesse modo, o serviço emite eventos de rastreamento de anúncios relacionados à reprodução, mas disponibiliza a carga útil completa de rastreamento do lado do cliente para a sessão

Para ativar o rastreamento híbrido usando prefixos de reprodução, a partir do player, inicialize uma nova sessão de MediaTailor reprodução usando uma solicitação em um dos seguintes formatos, de acordo com seu protocolo:

Example: formato HLS

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

Example: formato DASH

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

MediaTailor mantém os seguintes eventos de rastreamento no modo híbrido:

- Impressão
- Início
- Primeiro quartil
- Midpoint
- Terceiro quartil
- Concluído

- `breakStart(vmap)`
- `breakEnd(vmap)`

Client-side integrações de rastreamento de anúncios

Esta seção descreve as integrações entre MediaTailor e vários servidores de rastreamento de anúncios do lado do cliente.

Tópicos

- [SDK de medição aberto](#)
- [SDKs de reprodutores gratuitos do Datazoom](#)
- [Estrutura de publicidade Roku \(RAF\)](#)
- [TheoPlayer](#)
- [MediaTailor SDK](#)

SDK de medição aberto

O SDK de Medição Aberta (OM SDK) do Interactive Advertising Bureau (IAB) facilita a medição de visibilidade e verificação de anúncios veiculados em ambientes de vídeo na web e aplicativos nativos.

Para documentos mais antigos da versão 3 do VAST, o código de verificação deve ser carregado com o nó de extensão, com o tipo de extensão `AdVerifications`. A raiz do nó de extensão é um `AdVerifications` nó com o mesmo esquema do elemento VAST 4.1.

Para facilitar a adoção do SDK do OM, MediaTailor fez uma parceria com a Datazoom para fornecer SDKs gratuitos para jogadores que são configurados e verificados para o Open Measurement. Para obter mais informações, consulte [SDKs de reprodutores gratuitos do Datazoom](#).

Note

MediaTailor atualmente suporta apenas a versão 3 do VAST.

Example: nó de verificação no VAST 3, antes da versão 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 extrai os AdVerifications dados do <Extensions> nó e os coloca na adVerifications matriz na resposta de rastreamento do lado do cliente.

Example: matriz AdVerifications na resposta de rastreamento do lado do cliente

```

{
  "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",

```

[illegible]

```

        "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

Interaja com o IAB Tech Lab para garantir que os aplicativos sejam certificados anualmente para garantir a conformidade.

Para obter mais informações sobre o OM SDK, consulte [Open Measurement SDK no site](#) do IAB Tech Lab.

SDKs de reprodutores gratuitos do Datazoom

Para facilitar a adoção dos SDKs do player, MediaTailor fez uma parceria com a Datazoom para fornecer SDKs gratuitos para jogadores que são configurados e testados com o e o IAB Tech. [Client-side AWS Elemental MediaTailor integração com o Google Ad Manager SDK de medição aberto](#)

O SDK do Datazoom player é compatível com os seguintes recursos:

- Playlists ao vivo e VOD
- Especificações DASH e HLS
- Suporte do fornecedor do player para Bitmovin, exoplayer, reprodutor de mídia Android, Apple AVPlayer, Brightcove, Chromecast Receiver, hls.js, JWPlayer, Shaka player, THEO player Dash.js, Roku e muito mais Video.js
- Certificação de medição aberta do IAB Tech Lab, quando disponível em dispositivos selecionados
- Click-through tratamento de eventos
- Ad-event expedidores, como cronômetros de contagem regressiva de anúncios, eventos não lineares e de sobreposição de anúncios, início de intervalo de anúncio, fim de intervalo de anúncio
- Client-side e balizamento
- SDK da Biblioteca de Acesso Programático (PAL) do Google, como uma configuração opcional

O Datazoom também oferece um serviço pago de análise e telemetria compatível com os SDKs do player. Os clientes podem optar e controlar a telemetria do SDK do player a partir do console de gerenciamento do Datazoom. [Para acessar os SDKs do Datazoom player e saber mais sobre o serviço de telemetria e análise de valor agregado, use as informações de contato no site do Datazoom.](#)

Estrutura de publicidade Roku (RAF)

O Roku Ad Framework (RAF) mantém uma experiência publicitária consistente em toda a plataforma Roku. Todos os canais, incluindo anúncios em vídeo, devem atender aos requisitos de certificação da Roku para a RAF. Notavelmente, o aplicativo deve sempre usar o disparo de eventos do lado do cliente por meio do RAF. MediaTailor, como provedor de inserção de anúncios do lado do servidor (SSAI), oferece suporte ao disparo de eventos do lado do cliente. Os adaptadores RAFX SSAI fornecem interfaces tanto para servidores de manifestos SSAI, ou stitchers, quanto para RAF. Essas interfaces incluem:

- Análise e extração da masterURL resposta e playURL metadados AdURL do anúncio.
- Transformando metadados de anúncios MediaTailor SSAI em metadados de RAF-usable anúncios e configurando RAF para reprodução.
- Observando eventos de stream e metadados cronometrados.
- Combinando os eventos de transmissão, os metadados do anúncio e os pixels do evento de disparo em tempo hábil.

- Pinging/polling oAdURL, conforme exigido pelo servidor de manifesto MediaTailor SSAI, analisando e reconfigurando o RAF.

Para obter mais informações sobre adaptadores SSAI para RAF, consulte [Implementando a inserção de Server-Side anúncios usando adaptadores Roku no site da Roku](#).

TheoPlayer

TheoPlayer a integração com MediaTailor faz o seguinte:

- Fornece funcionalidade para oferecer suporte ao rastreamento de eventos MediaTailor do lado do cliente para HLS e DASH para VOD e fluxos de trabalho ao vivo.
- Suporta o envio de beacons de rastreamento somente para anúncios lineares.
- Desativa a busca durante um anúncio. No entanto, não há lógica para reproduzir um anúncio quando o usuário busca além do intervalo publicitário.

Para obter mais informações sobre a SSAI e analisar os SDKs para web, Android, iOS e tvOS MediaTailor, consulte [MediaTailor](#) no site. TheoPlayer TheoPlayer

MediaTailor SDK

AWS A Elemental mantém um kit de JavaScript-based desenvolvimento de software (SDK). AWS A Elemental fornece o SDK no estado em que se encontra, sem garantia implícita. Use o SDK como uma demonstração de referência para agilizar sua integração e uso. MediaTailor O SDK mostra como interagir com a API de rastreamento do MediaTailor lado do cliente. O SDK implementa o rastreamento e os relatórios de anúncios do lado do cliente para os jogadores. HTML5-based O SDK inicializa uma sessão de relatórios do MediaTailor lado do cliente e, em seguida, solicita periodicamente informações de rastreamento de anúncios. Durante a reprodução, o SDK emite eventos de rastreamento de anúncios quando novos eventos publicitários são detectados.

O MediaTailor SDK oferece suporte aos seguintes recursos:

- Playlists ao vivo e VOD
- Especificações DASH e HLS
- Click-through tratamento de eventos
- Ad-event despachantes
- Ganchos de eventos personalizados

- Client-side e faróis. Para obter mais informações sobre o envio de beacons de anúncios, consulte [Client-side balizamento](#)

Note

Envie um ticket de AWS Support para receber uma amostra de JavaScript SDK para MediaTailor. Você receberá um link para download do pacote e seus arquivos.

Paginando beacons de anúncios com GetTracking

Use o `GetTracking` endpoint para reduzir o número de anúncios retornados a um player. Por exemplo, se uma janela de manifesto for ampla, abrangendo muito tempo, o número de beacons de anúncios retornados pode afetar o desempenho do jogador.

`GetTracking` retorna um `NextToken` valor que você pode usar para restringir o número de beacons retornados paginando a lista de beacons retornados. Você pode percorrer os `NextToken` valores para encontrar o valor desejado do campo de um anúncio `StartTimeInSeconds` beacon.

- Na primeira chamada para `GetTracking`, todos os anúncios possíveis que aparecem na janela do manifesto são retornados, incluindo a `NextToken` e o valor de cada um.
- Se uma `GetTracking` solicitação não incluir um `NextToken`, todos os anúncios na janela do manifesto serão retornados.
- Se uma `GetTracking` solicitação contiver um `NextToken`, mas não houver novos beacons a serem retornados, MediaTailor retornará o mesmo valor `NextToken` que você enviou na solicitação original.
- Quando não há mais beacons correspondentes a um anúncio, `GetTracking` remove o anúncio de sua resposta.
- Os tokens `GetTracking` expiram após 24 horas. Se um `NextToken` valor tiver mais de 24 horas, a próxima chamada `GetTracking` retornará um valor nulo `NextToken`.

Sequência de chamada generalizada `GetTracking` do jogador

Do player cliente, uma `GetTracking` solicitação é um POST com um corpo de solicitação que contém anúncios `NextToken` e beacons relacionados ao token.


```
https://YouMediaTailorUrl/v1/tracking
{
    "NextToken": "value"
    .
    .
    .
}
```

A sequência geral para usar `GetTracking` com `NextToken` é a seguinte:

1. Faça a primeira ligação para `GetTracking`.

Todos os anúncios e beacons e os primeiros `NextToken` para chamadas subsequentes são retornados.

2. Se o valor de `NextToken` for nulo, MediaTailor retornará todos os beacons de anúncios.

3. Se o `NextToken` estiver expirado, MediaTailor retornará uma mensagem de erro do código de retorno HTTP 400.

Faça uma nova chamada `GetTracking` para recuperar `NextToken`s válidos.

4. Examine toda a resposta para encontrar um sinalizador `StartTimeInSeconds` de anúncios que esteja na faixa desejada.

5. Faça uma nova chamada para `GetTracking` com o valor de `NextToken` associado ao desejado `StartTimeInSeconds`.

6. Se necessário, percorra novamente os anúncios retornados até encontrar exatamente os que você deseja reproduzir.

Exemplo estendido

Este exemplo mostra como usar `GetTracking`'s `NextToken` para restringir o número de beacons de anúncios retornados a um jogador.

MediaTailor recebe uma `GetTracking` solicitação. A resposta contém um anúncio com ID 9935407 e dois beacons com `StartTimeInSeconds` valores 52.286 e 48.332 segundos.

MediaTailor envia a resposta JSON da `NextToken` seguinte forma:

```

{
  "NextToken": "JF57ITe48t1441mv7TmLKuZLroxDzfIslp6BiSNL1IJmzPVMdN0lqrBYycgMbKEb
  "avails": [
    {
      "ads": [
        {
          "adId": "9935407",
          "adVerifications": [],
          "companionAds": [],
          "creativeId": "",
          "creativeSequence": "",
          "duration": "PT15S",
          "durationInSeconds": 15,
          "extensions": [],
          "mediaFiles": {
            "mediaFilesList": [],
            "mezzanine": ""
          },
          "startTime": "PT30S",
          "StartTimeInSeconds": 45,
          "trackingEvents": [
            {
              "beaconUrls": [
                "http://adserver.com/tracking?event=Impression "
              ],
              "duration": "PT0S",
              "durationInSeconds": 0,
              "eventId": "9935414",
              "eventType": "secondQuartile",
              "startTime": "PT52.286S",
              "StartTimeInSeconds": 52.286
            },
            {
              "beaconUrls": [
                "http://adserver.com/tracking?event=firstQuartile"
              ],
              "duration": "PT0S",
              "durationInSeconds": 0,
              "eventId": "9935412",
              "eventType": "firstQuartile",
              "startTime": "PT48.332S",
              "StartTimeInSeconds": 48.332
            }
          ],
        },
      ],
    },
  ],
}

```

```

        "vastAdId": ""
      }
    ],
    "startTime": "PT46.47S",
    "StartTimeInSeconds": 46.47
  }
]
}

```

Na próxima `GetTracking` solicitação, MediaTailor responde com o `NextToken` valor: `JF57ITe48t1441mv7TmLKuZLroxDzflslp6BiSNL1IJmzPVMDN0lqrBYycgMbKEb`.

MediaTailor responde com anúncios e beacons que correspondem aos `StartTimeInSeconds` definidos na chamada `NextToken` anterior.

Suponha que agora a resposta inclua outro anúncio com ID 9235407, além do anúncio anterior com ID 9935407. Os beacons do anúncio ID 9235407 têm s 132.41 e 70.339. `StartTimeInSeconds`

MediaTailor itera sobre todos os beacons na sessão para selecionar aqueles com `StartTimeInSeconds` mais de 52.286 segundos, que são o beacon 3 e o beacon 4 do anúncio com ID 9235407:

```

{
  "NextToken": ZkfknbvbsdgfbbsDFRdffg12EdffecFRvhjyjfhdhfnjtsG5SDGN
  "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
}
]
}

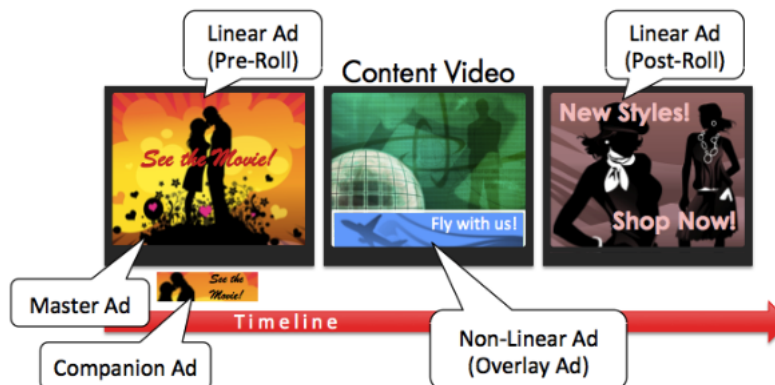
```

Anúncios de sobreposição

Para fluxos de trabalho de transmissão ao vivo nos quais você deseja aumentar a monetização sem interromper a experiência de visualização com anúncios intermediários, você pode aproveitar sua AWS Elemental MediaTailor integração atual para orientar um formato de publicidade renderizado no lado do cliente. Esse tipo de publicidade é conhecido como anúncios sobrepostos. Anúncios

de sobreposição são anúncios em vídeo não lineares que aparecem na forma de “anúncios”, “L-band anúncios em vídeo não lineares”, “anúncios de imagem em imagem”, “sobreposições em movimento”, “publicidade em conteúdo” ou “anúncios em quadro”.

MediaTailor detecta um SCTE-35 marcador com tipo de segmentação `id=0x38` como um sinal em banda para uma oportunidade de inserção de anúncios sobrepostos. O SCTE-35 marcador faz com MediaTailor que uma solicitação seja enviada ao Ad Decision Server (ADS), que então responde com uma carga de anúncio não linear na resposta VAST. MediaTailor analisa a resposta VAST para oferecer suporte à inserção de anúncios sobrepostos. MediaTailor não realiza nenhuma combinação de anúncios lineares, mas sinaliza para o player que há um anúncio de sobreposição não linear disponível para reprodução. Essa sinalização permite que o player busque e correlacione os anúncios não lineares a serem reproduzidos no endpoint de rastreamento do lado do cliente. O player então gerencia a exibição, os relatórios e outras tarefas relacionadas a esses anúncios. Por exemplo, o desenvolvedor do player pode usar um SDK de dispositivo de um fornecedor compatível com formatos de anúncios sobrepostos. Para obter mais informações sobre integrações de rastreamento do lado do cliente, consulte [Client-side integrações de rastreamento de anúncios](#)



Tópicos

- [Pré-requisitos para usar anúncios sobrepostos com MediaTailor](#)
- [Começando a usar anúncios de sobreposição com MediaTailor](#)
- [Registro e métricas para anúncios sobrepostos em MediaTailor](#)
- [Cobrança por anúncios sobrepostos em MediaTailor](#)

Pré-requisitos para usar anúncios sobrepostos com MediaTailor

Os pré-requisitos a seguir se aplicam ao usar anúncios de sobreposição com: MediaTailor

- O fluxo de trabalho deve ser ao vivo, não vídeo sob demanda (VOD).
- A resposta do Ad Decision Server (ADS) deve ser configurada para retornar somente anúncios não lineares na resposta VAST. MediaTailor ignora qualquer anúncio linear para fins de costura de anúncios.
- O manifesto deve usar uma mensagem de sinal de SCTE-35 horário com tipo de segmentação `id=0x38` para invocar o recurso de anúncio de sobreposição.
- O provedor de streaming deve ter controle do aplicativo cliente-dispositivo e estar integrado à API de rastreamento do lado do MediaTailor cliente.

Começando a usar anúncios de sobreposição com MediaTailor

Esta seção explica como começar a usar o recurso de anúncios sobrepostos do MediaTailor. Você configurará a sinalização SCTE-35, configurará as respostas do Ad Decision Server (ADS) e configurará o controle em nível de sessão.

Tópicos

- [Ativando anúncios de sobreposição](#)
- [Rastreamento de anúncios sobrepostos com metadados do lado do cliente](#)

Ativando anúncios de sobreposição

MediaTailor o suporte para anúncios de sobreposição está ativado por padrão. Um tipo específico de marcador de anúncio SCTE-35 no manifesto aciona a inserção de um anúncio de sobreposição. Como alguns players podem não oferecer suporte à renderização de anúncios sobrepostos no lado do cliente, você pode desativar o recurso no nível da sessão.

Para desativar o suporte a anúncios sobrepostos usando prefixos de reprodução HLS ou DASH:

- No player, inicialize uma nova sessão de MediaTailor reprodução usando uma solicitação em um dos seguintes formatos, de acordo com seu protocolo:

- Exemplo: formato HLS

```
GET mediatailorURL/v1/master/hashed-account-id/origin-id/asset-id?  
aws.overlayAvails=off
```

- Exemplo: formato DASH

```
GET mediatailorURL/v1/master/hashed-account-id/origin-id/asset-id?  
aws.overlayAvails=off
```

Para desativar o suporte a anúncios sobrepostos usando o prefixo de inicialização da sessão:

- No player, construa um corpo de mensagem JSON para a solicitação de inicialização da sessão para: MediaTailor
 - Para desativar o suporte à sobreposição de anúncios, adicione um `overlays` objeto como chave de nível superior com um valor de `off`. O `overlays` valor padrão é `on`.
 - (Opcional) Forneça quaisquer parâmetros que MediaTailor , em seguida, sejam passados para o ADS dentro de um `adsParams` objeto. Esses parâmetros correspondem às configurações `[player_params.param]` no URL de modelo ADS da configuração do MediaTailor.

Example HLS:

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

Example DASH:

```
POST manifest.mpd  
{  
  "adsParams": {  
    "deviceType": "androidmobile"  
  },  
  "overlayAvails": "off"  
}
```

Sinalização manifesta

MediaTailor aciona o suporte a anúncios de sobreposição quando vê um marcador SCTE-35 específico no manifesto. O sinal necessário é um comando de emenda tipo 6, ou sinal de tempo, que é um sinal Provider Overlay Advertisement Start. Este sinal tem um ID do tipo de segmentação de 0x38

O exemplo a seguir mostra o marcador 0x38 SCTE-35 em um objeto JSON.

```
{
  "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
}

```

O exemplo a seguir mostra o sinal SCTE-35 representado como um valor binário (base 32/hexadecimal):

```
0xfc303500000000000000ff00506fe001b78c8001f021d435545490970d4717fdf0000dbba009076f7665726c617
```

Os exemplos a seguir mostram o marcador SCTE-35 nos manifestos HLS e DASH.

Example: Manifesto HLS

```
#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,
```

```
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:25.251Z
#EXTINF:2.0,
https://aws.cloudfront.net/media/asset1/index1_00002.ts
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:27.251Z
#EXTINF:6.0,
https://aws.cloudfront.net/media/asset1/index1_00003.ts
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:33.251Z
#EXTINF:2.0,
https://aws.cloudfront.net/media/asset1/index1_00004.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_00005.ts
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:39.251Z
#EXTINF:6.02,
https://aws.cloudfront.net/media/asset1/index1_00006.ts
#EXT-X-DISCONTINUITY
#EXT-X-PROGRAM-DATE-TIME:2023-08-15T04:30:41.251Z
#EXTINF:6.02,
https://aws.cloudfront.net/media/asset1/index1_00007.ts
```

Example: Manifesto DASH

```
<?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>

```

Resposta do Ad Decision Server (ADS)

A resposta do ADS deve conter um evento de rastreamento válido. No mínimo, o evento de rastreamento pode ser um evento `Impression` de rastreamento. O evento de rastreamento deve conter pelo menos um `NonLinear` anúncio. Esse anúncio é o anúncio de sobreposição, assumindo a forma de um recurso estático, `HTML` ou `iFrame`.

```
<vmap AdBreak breaktype="linear" breakId="csoverlay"
```

Se a resposta VAST for um VMAP com breakType ofnonlinear, os metadados disponíveis estarão dentro do nonLinearAvails objeto raiz. Se a resposta VAST for um VMAP com um breakType de linear ou for uma resposta VAST simples sem VMAP, os metadados disponíveis estarão dentro do objeto raiz. avails

A resposta VAST a seguir é uma resposta VMAP encapsulada com um breakType valor de linear.

Além da resposta VMAP encapsulada, MediaTailor também suporta uma resposta VMAP encapsulada com um breakType valor de nonlinear, e uma resposta VAST simples.

```
<?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:TrackingEvents>
  </vmap:AdBreak>
</vmap:VMAP>
```

```

    <vmap:Tracking event="breakEnd"><![CDATA[https://adserver.com/
beacon=breakendimpression]]></vmap:Tracking>
  </vmap:TrackingEvents>
</vmap:AdBreak>
</vmap:VMAP>

```

Example 1: Fonte do manifesto DASH para MediaTailor

```

<?xml version="1.0" encoding="utf-8"?>
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd" id="201"
  type="dynamic" publishTime="2022-11-07T19:59:05+00:00" minimumUpdatePeriod="PT2S"
  availabilityStartTime="2022-11-07T06:57:11.250000+00:00" minBufferTime="PT10S"
  suggestedPresentationDelay="PT20.000S" timeShiftBufferDepth="PT58.999S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011">
  <Period start="PT46827.601S" id="0" duration="PT88.321S">
    ...
  </Period>
  <Period start="PT46915.922S" id="45" duration="PT6.006S">
    <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2014:xml+bin">
      <Event duration="540000" id="144">
        <scte35:Signal>
          <scte35:Binary>SCTE35-binary</scte35:Binary>
        </scte35:Signal>
      </Event>
    </EventStream>
    ...
  </Period>
  <Period start="PT46921.928S" id="49">
    ...
  </Period>
</MPD>

```

Example 2: manifesto DASH MediaTailor personalizado contendo uma decoração de ID de anúncio

```

<?xml version="1.0" encoding="utf-8"?>
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd" id="201"
  type="dynamic" publishTime="2022-11-07T19:59:05+00:00" minimumUpdatePeriod="PT2S"

```

```

availabilityStartTime="2022-11-07T06:57:11.250000+00:00" minBufferTime="PT10S"
suggestedPresentationDelay="PT20.000S" timeShiftBufferDepth="PT58.999S"
profiles="urn:mpeg:dash:profile:isoff-live:2011">
  <Period start="PT46827.601S" id="0" duration="PT88.321S">
    ...
  </Period>
  <Period start="PT46915.922S" id="45" duration="PT6.006S">
    <EventStream schemeIdUri="urn:sva:advertising-wg:ad-id-signaling" timescale="90000">
      <Event presentationTime="13500000" duration="1351350">
        <![CDATA[{"version": 1,"identifiers": [{"scheme":
"urn:smp:ul:060E2B34.01040101.01200900.00000000","value": "adId","ad_position":
"adId", "ad_type":"overlay","creative_id": "creativeId","tracking_uri":
"trackingUri"}]]]]></Event>
      </EventStream>
    ...
  </Period>
  <Period start="PT46921.928S" id="49">
    ...
  </Period>
</MPD>

```

Rastreamento de anúncios sobrepostos com metadados do lado do cliente

MediaTailor coloca os anúncios de sobreposição no local `nonLinearAdsList` da disponibilidade. A API de rastreamento MediaTailor do lado do cliente tem dois objetos raiz, chamados `e` e `avails`. Se a resposta VAST for um VMAP com `breakType` de `ofnonlinear`, os metadados disponíveis estarão dentro do `nonLinearAvails` objeto raiz. Se a resposta VAST for um VMAP com um `breakType` de `linear` ou for uma resposta VAST simples sem VMAP, os metadados disponíveis estarão dentro do objeto raiz `avails`.

Para obter mais informações sobre o rastreamento do lado do cliente, consulte [Client-side rastreamento de anúncios](#).

O exemplo a seguir mostra uma resposta VAST simples ou uma resposta VMAP com um `breakType` valor de `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": []
}

```

O exemplo a seguir mostra uma resposta simples do VMAP com um breakType valor denonlinear.

```

{
    "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
}
]
}

```

Registro e métricas para anúncios sobrepostos em MediaTailor

Esta seção explica o registro e as métricas para anúncios sobrepostos em MediaTailor. Para obter mais informações sobre como configurar o registro, consulte [Monitorar e atribuir tags \(tagging\) AWS Elemental MediaTailor recursos](#).

Tópicos

- [CloudWatch troncos](#)
- [CloudWatch métricas](#)

CloudWatch troncos

CloudWatch coleta as seguintes informações de registro sobre anúncios de sobreposição:

- **VAST_RESPONSE**- Mostra informações sobre a lista de anúncios não lineares.
- **FILLED_PROVIDER_OVERLAY**- Mostra informações sobre os anúncios não lineares.

Note

RAW_ADS_RESPONSE é um evento opcional que mostra a resposta original do ADS. Usar esse evento é especialmente útil em um ambiente de preparação e teste. Para ativar esse evento em uma configuração ou conta, envie um ticket para o AWS Support.

CloudWatch métricas

MediaTailor coleta métricas de anúncios de sobreposição separadamente de outras métricas do ADS. MediaTailor coleta essas métricas depois de obter com sucesso os anúncios do ADS. Você não precisa pesquisar a `GetTracking` API para coletar as métricas.

A tabela a seguir descreve CloudWatch as métricas para anúncios sobrepostos:

Métrica	Description
<code>AdDecisionServer.OverlayAds</code>	A contagem de anúncios sobrepostos incluídos nas respostas do ADS dentro do CloudWatch período que você especificou.
<code>AdDecisionServer.OverlayErrors</code>	O número de respostas de código de 200 status não HTTP, respostas vazias e respostas com tempo limite MediaTailor recebidas do ADS dentro do CloudWatch período especificado.
<code>AdDecisionServer.OverlayFilled</code>	O número de disponibilidades que foram preenchidas com sucesso com pelo menos um anúncio de sobreposição: • 1 - Há pelo menos um anúncio válido.

Métrica	Description
	0 - Ou MediaTailor não recebi nenhum anúncio de sobreposição ou houve alguma outra falha. SampleCount rastreia o número de disponibilidades preenchidas. Sum rastreia o número de sobreposições preenchidas com sucesso.
AdDecisionServer.OverlayMinSuggestedDuration	A soma das minSuggestedDuration durações, em milissegundos, de todos os anúncios MediaTailor recebidos do ADS dentro do CloudWatch período que você especificou. Se minSuggestedDuration não for especificada, a duração mostrada será a duração planejada.
AdDecisionServer.OverlayLatency	O tempo de resposta, em milissegundos, para solicitações feitas MediaTailor ao ADS.
AdDecisionServer.OverlayTimeouts	O número de solicitações expiradas para o ADS no CloudWatch período que você especificou.
AdsBilled	Para obter mais informações sobre anúncios cobrados, consulte Cobrança por anúncios sobrepostos em MediaTailor .
Avail.*	Porque MediaTailor não faz nenhum planejamento para anúncios sobrepostos, CloudWatch não mostra nenhuma Avail.X métrica.

Métrica	Description
SkippedReason.*	Porque MediaTailor não faz nenhum planejamento para anúncios sobrepostos, CloudWatch não mostra nenhuma SkippedReason.X métrica.

Cobrança por anúncios sobrepostos em MediaTailor

MediaTailor cobra dos clientes com base no número de anúncios não lineares na resposta do ADS. Esse número inclui anúncios não lineares que se estendem além da duração do intervalo. Depois de MediaTailor preencher o formulário, ele cobra pelos anúncios preenchidos.

Para fluxos de trabalho de pré-busca, MediaTailor não cobra pelos anúncios ao recuperar a pré-busca, mas sim quando vê um anúncio compatível disponível na janela de consumo dessa sessão.

Para obter informações adicionais sobre faturamento, consulte <https://aws.amazon.com/mediatailor/pricing/>.

Decoração de ID do anúncio

AWS Elemental MediaTailor realiza a junção de anúncios do lado do servidor ao fazer a transição do conteúdo para os intervalos publicitários. MediaTailor pode condicionar o manifesto com metadados associados aos anúncios que foram criados. Fazer isso pode oferecer os seguintes benefícios:

Important

A decoração do Ad-ID não é compatível com os métodos de inserção de anúncios guiada pelo servidor (SGAI) porque os campos que preenchem os X-AD-CREATIVE-SIGNALING cabeçalhos só são conhecidos quando a lista de ativos é buscada durante a reprodução do anúncio, não quando o manifesto que pode ser armazenado em cache é gravado inicialmente.

- O horário de início do vídeo (VST) melhora
- MediaTailor pode suportar um modelo híbrido de inserção de anúncios do lado do servidor e inserção de anúncios guiada pelo servidor

- As sessões do lado do servidor podem criar cronogramas de reprodução com marcadores de posição do anúncio
- Para sessões do lado do cliente que já criam cronogramas de reprodução com a MediaTailor API, o VST da sessão melhora, pois a sessão não depende da chamada da API de rastreamento para criar a linha do tempo
- É possível aproveitar a inserção MediaTailor de anúncios do lado do servidor, bem como os anúncios renderizados do lado do cliente exibidos na cena. Dessa forma, o kit de desenvolvimento de software (SDK) de um player não precisa ter uma integração separada para chamar entidades de veiculação de anúncios diretamente para anúncios do lado do cliente. MediaTailor pode vender os anúncios por meio do manifesto e da API de rastreamento do lado do cliente.

Existem padrões para associar cada recurso de anúncio criativo a um identificador exclusivo. Essa associação permite que anunciantes, agências, fornecedores e editores relacionem um ativo publicitário criativo em seus fluxos de trabalho independentes. À medida que as métricas e o monitoramento dos fluxos continuam melhorando e mais distribuidores utilizam arquiteturas de inserção baseadas em servidor, surge a necessidade de comunicar com precisão os identificadores atribuídos aos ativos criativos individuais em uma interleaved/stitched apresentação, como no manifesto personalizado.

Tópicos

- [Habilitando a sinalização de ID de anúncio para sessões](#)
- [Manifestos e inserção de metadados de anúncios](#)
- [Interações do Ad Decision Server \(ADS\)](#)
- [API de rastreamento do lado do cliente](#)

Habilitando a sinalização de ID de anúncio para sessões

O recurso de sinalização de ID de anúncio deve ser ativado durante a inicialização da sessão. O processo para ativar o recurso é diferente de criar sessões usando o prefixo de HLS/DASH reprodução (inicialização implícita da sessão) versus o prefixo de inicialização da sessão (inicialização explícita da sessão).

Para ativar o ID de anúncio para a sessão usando prefixos de HLS/DASH reprodução

- No player, inicialize uma nova sessão de MediaTailor reprodução usando uma solicitação em um dos seguintes formatos, de acordo com seu protocolo:

- Exemplo: formato HLS

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

- Exemplo: formato DASH

```
GET <mediatailorURL>/v1/dash/<hashed-account-id>/<origin-id>/<asset-id>?
aws.adSignalingEnabled=true
```

Para habilitar o ID de anúncio para a sessão usando o prefixo de inicialização da sessão

- No player, construa um corpo de mensagem JSON para a solicitação de inicialização da sessão para: MediaTailor
 - Dentro de um adsParams objeto, forneça todos os parâmetros que MediaTailor devem ser passados para o ADS. Esses parâmetros correspondem às [player_params.param] configurações na URL do modelo ADS da MediaTailor configuração.
 - Para ativar a sinalização de ID de anúncio, adicione um adSignaling objeto como objeto de nível superior e, dentro, adicione um parâmetro chamado enabled e valor de. true O adSignaling valor padrão é disabled.
- Exemplo: formato HLS

```
POST master.m3u8
{
  "adsParams": {
    "deviceType": "ipad"
  },
  "adSignaling": {
    "enabled": "true"
  },
  "reportingMode": "client"
}
```

- Exemplo: formato DASH

```
POST manifest.mpd
```



```
{
  "adsParams": {
    "deviceType": "ipad"
  },
  "adSignaling": {
    "enabled": "true"
  },
  "reportingMode": "client"
}
```

Manifestos e inserção de metadados de anúncios

Durante o processo de criação do anúncio, MediaTailor adiciona ao manifesto o ID exclusivo associado a cada criação que está sendo costurada. MediaTailor obtém a ID exclusiva do criativo a partir do valor do `id` atributo desse criativo na resposta VAST. Se o criativo não tiver um valor de atributo de ID, MediaTailor publicará um valor vazio (`id=""`).

MediaTailor usa um sinal de metadados no manifesto para separar as dependências entre a API de rastreamento do cliente para metadados criativos de anúncios e dentro do cronograma geral. `timing/positioning` Esse desacoplamento reduz a latência da reprodução (especialmente em cenários de VOD), em que a interface do usuário (UI) do player renderiza as posições de interrupção do anúncio na linha do tempo antes de inicializar a reprodução.

Os metadados adicionados assumem as seguintes formas:

- Para manifestos HLS, os metadados adicionados assumem a forma de `DATERANGE` tags para cada anúncio no período de disponibilidade.
- Para manifestos do DASH, os metadados adicionados assumem a forma de um `Event` elemento dentro de cada período do anúncio.

O corpo da mensagem JSON a seguir mostra um exemplo de resposta VAST:

```
{
  "version": 1,
  "identifiers": [
    {
      "scheme": "urn:smp:ul:060E2B34.01040101.01200900.00000000",
      "value": "creativeId",
      "ad_position": "adId",
    }
  ]
}
```

```

    "ad_type": "adType",
    "tracking_uri": "trackingUri",
    "custom_vast_data": "customVastData"
  }
]
}

```

No exemplo anterior:

- *creativeId* é o valor do Id atributo do Creative elemento para o anúncio
- *adId* é o número de sequência HLS associado ao início do anúncio ou o ID do período DASH do anúncio
- *adType* é *avail* ou *overlay*, com base na resposta VAST
- *trackingUri* é o endpoint de rastreamento relativo da MediaTailor sessão, no formato `../../../../tracking/hashed-account-id/origin-id/session-id`
- *customVastData* é um valor MediaTailor extraído da extensão `creative_signaling` VAST. MediaTailor usa o conteúdo do nó CDATA, se presente. Consulte a [Interações do Ad Decision Server \(ADS\)](#) seção para obter mais detalhes e um exemplo de resposta VAST.

Personalização de manifestos HLS com metadados de anúncios

Para um stream HLS ao vivo, MediaTailor só adiciona metadados quando o stream contém PROGRAM-DATE-TIME tags, pelo menos uma vez por duração do manifesto. Para um stream de vídeo sob demanda (VOD), é adicionado PROGRAM-DATE-TIME a pelo menos um segmento no manifesto personalizado, em que a hora de início de cada ativo de VOD é a época zero (). 1970-01-01T00:00:00Z Se o manifesto de origem tiver PROGRAM-DATE-TIME conteúdo existente, MediaTailor preservará esse conteúdo.

MediaTailor personaliza o manifesto com criativos retornados pelo Ad Decision Server (ADS). Para cada anúncio, MediaTailor também inclui uma DATERANGE tag que abrange a duração do anúncio. O formato da DATERANGE tag é semelhante ao descrito na seção [Sinalização criativa de anúncios em DASH e HLS](#) na versão 2023 da publicação técnica do SVA.

O DATERANGE que MediaTailor gera tem valores de ID exclusivos. Para garantir a exclusividade (de acordo com as diretrizes especificadas em [Mapeamento do SCTE-35 em EXT-X-DATERANGE](#)) MediaTailor, associa o MEDIA-SEQUENCE número do primeiro segmento do anúncio do item ao número de sequência do anúncio dentro do item.

Para intervalos publicitários não preenchidos em configurações que têm o slate ativado, MediaTailor anexa os segmentos do slate ao final da disponibilidade, separados por uma DISCONTINUITY tag, mas sem metadados. DATERANGE

Para cada anúncio incluído no manifesto personalizado, MediaTailor adiciona os metadados criativos, representados como dados codificados em base64 em uma tag personalizada. DATERANGE

Example Origem linear do 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=/DA1AAAAAyYAP/wFAUAAAACf
+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_406.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=42.175,Duration=59.993,SCTE35=/DA1AAAAAyYAP/wFAUAAAACf
+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_407.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=48.181,Duration=59.993,SCTE35=/DA1AAAAAyYAP/wFAUAAAACf
+//jP197P4AUmNiAAEBAQAase4/gA==
#EXTINF:6.006,
index_1_408.ts?m=1676054627
#EXT-X-CUE-OUT-CONT:ElapsedTime=54.187,Duration=59.993,SCTE35=/DA1AAAAAyYAP/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 Origem linear do HLS (**#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 Manifesto HLS linear personalizado (com sinalização criativa de anúncios):

O DATERANGE que MediaTailor gera tem valores de ID exclusivos. Para garantir a exclusividade (de acordo com as diretrizes especificadas em [Mapeamento do SCTE-35 em EXT-X-DATERANGE](#)) MediaTailor , associa o MEDIA-SEQUENCE número do primeiro segmento do anúncio do item ao número de sequência do anúncio dentro do item.

No exemplo a seguir, MediaTailor concatena MEDIA-SEQUENCE 421 com o número da posição do anúncio.

```

#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 Origem do VOD HLS (com sinais SCTE):

```
#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 Origem do VOD HLS:

```
#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 Manifesto personalizado VOD HLS:

MediaTailor adiciona PROGRAM-DATE-TIME aos manifestos de VOD para usá-los como âncoras para os DATERANGE elementos HLS que indicam as posições do anúncio.

O DATERANGE que MediaTailor gera tem valores de ID exclusivos. Para garantir a exclusividade (de acordo com as diretrizes especificadas em [Mapeamento do SCTE-35 em EXT-X-DATERANGE](#)) MediaTailor , associa o MEDIA-SEQUENCE número do primeiro segmento do anúncio do item ao número de sequência do anúncio dentro do item.

No exemplo a seguir, MediaTailor concatena MEDIA-SEQUENCE 421 com o número da posição do anúncio.

```
#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/cunsc-media/SK0-22/asset-1/hls/
index_720p1500k_00011.ts
#EXTINF:6.0,
https://d3fch9e2fcarly.cloudfront.net/cunsc-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
```

Personalização de manifestos do DASH com metadados de anúncios

MediaTailor personaliza o manifesto com criativos retornados pelo Ad Decision Server (ADS). Para cada anúncio, MediaTailor também inclui um `EventStream` elemento que abrange a duração do anúncio. O formato do *Event* elemento é semelhante ao descrito na seção [Sinalização criativa de anúncios em DASH e HLS](#) na versão 2023 da publicação técnica do SVA.

Para intervalos publicitários não preenchidos em configurações que têm o Slate ativado, MediaTailor acrescenta o período do Slate ao final do período de disponibilidade, mas sem nenhum metadado `EventStream`

Para cada anúncio incluído no manifesto personalizado, MediaTailor adiciona os metadados criativos, representados como um `CDATA` elemento dentro de um elemento `Event`

Exemplo Origem linear do DASH (atributos SCTE em linha):

```
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:scte35="urn:scte:scte35:2013:xml"
  xsi:schemaLocation="urn:mpeg:dash:schema:mpd:2011 http://standards.iso.org/
  ittf/PubliclyAvailableStandards/MPEG-DASH_schema_files/DASH-MPD.xsd" id="201"
  type="dynamic" publishTime="2023-02-10T21:08:40+00:00" minimumUpdatePeriod="PT6S"
  availabilityStartTime="2023-02-09T22:47:05.865000+00:00" minBufferTime="PT10S"
  suggestedPresentationDelay="PT20.000S" timeShiftBufferDepth="PT88.999S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011">
  <Period start="PT80141.456S" id="104" duration="PT304.103S">
    <AdaptationSet id="1485523442" mimeType="video/mp4" segmentAlignment="true"
    startWithSAP="1" subsegmentAlignment="true" subsegmentStartsWithSAP="1"
    bitstreamSwitching="true">
      <SegmentTemplate timescale="60000" media="index_video_${RepresentationID$_0_
      $Number$.mp4?m=1676062374" initialization="index_video_${RepresentationID$_0_init.mp4?
      m=1676062374" startNumber="151" presentationTimeOffset="4808487386">
        <SegmentTimeline>
```

```

        <S t="4824975858" d="360360" r="3"/>
        <S t="4826417298" d="316316"/>
    </SegmentTimeline>
</SegmentTemplate>
<Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="1800000" codecs="avc1.4D401F"/>
    <Representation id="3" width="640" height="360" frameRate="30000/1001"
bandwidth="1200000" codecs="avc1.4D401E"/>
    <Representation id="5" width="480" height="270" frameRate="30000/1001"
bandwidth="800000" codecs="avc1.4D4015"/>
</AdaptationSet>
<AdaptationSet id="1377232898" mimeType="audio/mp4" segmentAlignment="0"
lang="eng">
    <Label>eng</Label>
    <SegmentTemplate timescale="48000" media="index_audio_$RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_audio_$RepresentationID$_0_init.mp4?
m=1676062374" startNumber="151" presentationTimeOffset="3846790126">
        <SegmentTimeline>
            <S t="3859981294" d="287744"/>
            <S t="3860269038" d="288768"/>
            <S t="3860557806" d="287744"/>
            <S t="3860845550" d="288768"/>
            <S t="3861134318" d="252928"/>
        </SegmentTimeline>
    </SegmentTemplate>
    <Representation id="2" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
    <Representation id="4" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
    <Representation id="6" bandwidth="193007" audioSamplingRate="48000"
codecs="mp4a.40.2">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
</AdaptationSet>
<SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2023-02-10T21:02:31.007Z"/>
</Period>

```

```

<Period start="PT80445.560S" id="155" duration="PT44.978S">
  <EventStream timescale="90000" schemeIdUri="urn:scte:scte35:2013:xml">
    <Event duration="4048044">
      <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="207000"
tier="4095">
        <scte35:SpliceInsert spliceEventId="111" spliceEventCancelIndicator="false"
outOfNetworkIndicator="true" spliceImmediateFlag="false" uniqueProgramId="1"
availNum="1" availsExpected="1">
          <scte35:Program>
            <scte35:SpliceTime ptsTime="7239893422"/>
          </scte35:Program>
          <scte35:BreakDuration autoReturn="true" duration="4048044"/>
        </scte35:SpliceInsert>
      </scte35:SpliceInfoSection>
    </Event>
  </EventStream>
  <AdaptationSet id="1485523442" mimeType="video/mp4" segmentAlignment="true"
startWithSAP="1" subsegmentAlignment="true" subsegmentStartsWithSAP="1"
bitstreamSwitching="true">
    <SegmentTemplate timescale="60000" media="index_video_$RepresentationID$_0_
$Number$.mp4?m=1676062374" initialization="index_video_$RepresentationID$_0_init.mp4?
m=1676062374" startNumber="156" presentationTimeOffset="4826733614">
      <SegmentTimeline>
        <S t="4826733614" d="284284"/>
        <S t="4827017898" d="360360" r="5"/>
        <S t="4829180058" d="252252"/>
      </SegmentTimeline>
    </SegmentTemplate>
    <Representation id="1" width="960" height="540" frameRate="30000/1001"
bandwidth="1800000" codecs="avc1.4D401F"/>
    <Representation id="3" width="640" height="360" frameRate="30000/1001"
bandwidth="1200000" codecs="avc1.4D401E"/>
    <Representation id="5" width="480" height="270" frameRate="30000/1001"
bandwidth="800000" codecs="avc1.4D4015"/>
  </AdaptationSet>
  <AdaptationSet id="1377232898" mimeType="audio/mp4" segmentAlignment="0"
lang="eng">
    <Label>eng</Label>
    <SegmentTemplate timescale="48000" media="index_audio_$RepresentationID$_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"/>
      </SegmentTimeline>
    </SegmentTemplate>
  </AdaptationSet>
</Period>

```

```

        <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"/>
    </AdaptationSet>
</Period>

```



```

    <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 Manifesto personalizado linear do DASH (com sinalização criativa de anúncios):

```

<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/
f9f38deca3f14fc4b5ab3cddb76cfb9e/</BaseURL>
    <Location>https://777788889999.mediatailor.us-west-2.amazonaws.com/
v1/dash/94063eadf7d8c56e9e2edd84fdf897826a70d0df/emt/out/v1/
f9f38deca3f14fc4b5ab3cddb76cfb9e/index.mpd?
aws.sessionId=672ed481-4ffd-4270-936f-7c8403947f2e</Location>
    <Period duration="PT304.103S" id="104" start="PT80141.456S">
        <AdaptationSet bitstreamSwitching="true" id="1485523442" mimeType="video/
mp4" segmentAlignment="true" startWithSAP="1" subsegmentAlignment="true"
subsegmentStartsWithSAP="1">
            <SegmentTemplate initialization="index_video_$RepresentationID$_init.mp4?
m=1676062374" media="index_video_$RepresentationID$_$Number$.mp4?m=1676062374"
presentationTimeOffset="4808487386" startNumber="151" timescale="60000">
                <SegmentTimeline>
                    <S d="360360" r="3" t="4824975858"/>
                    <S d="316316" t="4826417298"/>
                </SegmentTimeline>
            </SegmentTemplate>
            <Representation bandwidth="1800000" codecs="avc1.4D401F"
frameRate="30000/1001" height="540" id="1" width="960"/>
            <Representation bandwidth="1200000" codecs="avc1.4D401E"
frameRate="30000/1001" height="360" id="3" width="640"/>
            <Representation bandwidth="800000" codecs="avc1.4D4015"
frameRate="30000/1001" height="270" id="5" width="480"/>
        </AdaptationSet>
        <AdaptationSet id="1377232898" lang="eng" mimeType="audio/mp4"
segmentAlignment="0">
            <Label>eng</Label>
            <SegmentTemplate initialization="index_audio_$RepresentationID$_init.mp4?
m=1676062374" media="index_audio_$RepresentationID$_$Number$.mp4?m=1676062374"
presentationTimeOffset="3846790126" startNumber="151" timescale="48000">
                <SegmentTimeline>
                    <S d="287744" t="3859981294"/>
                    <S d="288768" t="3860269038"/>
                    <S d="287744" t="3860557806"/>
                    <S d="288768" t="3860845550"/>
                    <S d="252928" t="3861134318"/>
                </SegmentTimeline>
            </SegmentTemplate>
            <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="2">

```

```

        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
    <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="4">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
    <Representation audioSamplingRate="48000" bandwidth="193007"
codecs="mp4a.40.2" id="6">
        <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
</AdaptationSet>
    <SupplementalProperty schemeIdUri="urn:scte:dash:utc-time"
value="2023-02-10T21:02:31.007Z"/>
</Period>
    <Period id="155_1" start="PT22H20M45.56S">
        <BaseURL>https://777788889999.mediatailor.us-west-2.amazonaws.com/
v1/dashsegment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/
emt/672ed481-4ffd-4270-936f-7c8403947f2e/155/155_1/</BaseURL>
        <EventStream schemeIdUri="urn:sva:advertising-wg:ad-id-signaling"
timescale="90000">
            <Event presentationTime="xxxxx" duration="1351350">
                <![CDATA[{"version": 1,"identifiers": [{"scheme":
"urn:smppte:ul:060E2B34.01040101.01200900.000000000", "value": "155_1", "ad_position":
"155_1", "ad_type": "avail", "creative_id": "123", "tracking_uri": "../..../v1/
tracking/hashed-account-id/origin-id/session-id", "custom_vast_data": "123abc"}]]]>
            </Event>
        </EventStream>
        <AdaptationSet bitstreamSwitching="false" frameRate="30000/1001"
mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
subsegmentAlignment="true" subsegmentStartsWithSAP="1">
            <SegmentTemplate startNumber="1" timescale="90000"/>
            <Representation bandwidth="1800000" codecs="avc1.64001f" height="540"
id="1" width="960">
                <SegmentTemplate initialization="asset_540_1_2init.mp4"
media="asset_540_1_2_${Number%09d$.mp4" startNumber="1" timescale="90000">
                    <SegmentTimeline>
                        <S d="180180" r="6" t="0"/>
                        <S d="90090" t="1261260"/>
                    </SegmentTimeline>
                </SegmentTemplate>
            </Representation>

```

```

    <Representation bandwidth="1200000" codecs="avc1.64001e" height="360"
id="3" width="640">
      <SegmentTemplate initialization="asset_360_1_1init.mp4"
media="asset_360_1_1_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180180" r="6" t="0"/>
          <S d="90090" t="1261260"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
    <Representation bandwidth="800000" codecs="avc1.640015" height="270" id="5"
width="480">
      <SegmentTemplate initialization="asset_270_0_0init.mp4"
media="asset_270_0_0_${Number%09d$.mp4" startNumber="1" timescale="90000">
        <SegmentTimeline>
          <S d="180180" r="6" t="0"/>
          <S d="90090" t="1261260"/>
        </SegmentTimeline>
      </SegmentTemplate>
    </Representation>
  </AdaptationSet>
  <AdaptationSet lang="eng" mimeType="audio/mp4" segmentAlignment="0">
    <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000"/>
    <Label>eng</Label>
    <Representation audioSamplingRate="48000" bandwidth="128000"
codecs="mp4a.40.2" id="6">
      <SegmentTemplate initialization="asset_audio_128_3init.mp4"
media="asset_audio_128_3_${Number%09d$.mp4" startNumber="1" timescale="48000">
        <SegmentTimeline>
          <S d="98304" t="0"/>
          <S d="96256" r="1" t="98304"/>
          <S d="95232" t="290816"/>
          <S d="96256" r="2" t="386048"/>
          <S d="48128" t="674816"/>
        </SegmentTimeline>
      </SegmentTemplate>
    <AudioChannelConfiguration
schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
    </Representation>
  </AdaptationSet>
</Period>
<Period id="155_2" start="PT22H21M0.575S">

```

```

    <BaseURL>https://777788889999.mediatailor.us-west-2.amazonaws.com/
v1/dashsegment/94063eadf7d8c56e9e2edd84fdf897826a70d0df/
emt/672ed481-4ffd-4270-936f-7c8403947f2e/155/155_2/</BaseURL>
    <EventStream schemeIdUri="urn:sva:advertising-wg:ad-id-signaling"
timescale="90000">
        <Event presentationTime="0" duration="1351350">
            <![CDATA[{"version": 1,"identifiers": [{"scheme":
"urn:smppte:ul:060E2B34.01040101.01200900.00000000","value": "155_2","ad_position":
"155_2", "ad_type":"avail","creative_id": "234","tracking_uri": "../v1/
tracking/hashed-account-id/origin-id/session-id","custom_vast_data":"123abc"}]]]>
        </Event>
    </EventStream>
    <AdaptationSet bitstreamSwitching="false" frameRate="30000/1001"
mimeType="video/mp4" segmentAlignment="true" startWithSAP="1"
subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SegmentTemplate startNumber="1" timescale="90000"/>
        <Representation bandwidth="1800000" codecs="avc1.64001f" height="540"
id="1" width="960">
            <SegmentTemplate initialization="asset_540_1_2init.mp4"
media="asset_540_1_2_${Number%09d$.mp4}" startNumber="1" timescale="90000">
                <SegmentTimeline>
                    <S d="180180" r="6" t="0"/>
                    <S d="90090" t="1261260"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <Representation bandwidth="1200000" codecs="avc1.64001e" height="360"
id="3" width="640">
            <SegmentTemplate initialization="asset_360_1_1init.mp4"
media="asset_360_1_1_${Number%09d$.mp4}" startNumber="1" timescale="90000">
                <SegmentTimeline>
                    <S d="180180" r="6" t="0"/>
                    <S d="90090" t="1261260"/>
                </SegmentTimeline>
            </SegmentTemplate>
        </Representation>
        <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="155_3" start="PT22H21M15.59S">
    <BaseURL>https://777788889999.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>
<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>

```

Interações do Ad Decision Server (ADS)

MediaTailor usa o valor do `id` atributo criativo da resposta VAST como um valor na sinalização do ID do anúncio. Se o valor do `id` atributo estiver vazio ou não estiver presente na resposta VAST, MediaTailor insere um valor vazio na sinalização do ID do anúncio.

Example Resposta VAST:

O exemplo de resposta VAST a seguir inclui um valor de `id` atributo para o linear Creative em linha. MediaTailor extrai o valor do Extension elemento VAST personalizado e coloca esse valor nos metadados criativos do manifesto.

```

<?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>

```

API de rastreamento do lado do cliente

O exemplo a seguir mostra como o SDK de um player vincula os metadados do anúncio no manifesto aos dados completos do evento de rastreamento na carga útil de resposta de rastreamento do lado do cliente com e. `creativeId` `adId`

Example Mensagem JSON:

```

{
  "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": "UFQ1TTM0Ljk2N1NfMjAyMi0xMS0xOFQwNDZMzo1Mi4yNDUxOTdaXzE%3D",
"nonLinearAvails": []
}
```

Usando AWS Elemental MediaTailor para criar fluxos montados lineares

AWS Elemental MediaTailor a montagem de canais é um serviço exclusivo para manifestos que permite criar canais de streaming linear usando seu conteúdo de vídeo sob demanda (VOD) existente misturado com conteúdo ao vivo. MediaTailor nunca toca em seus segmentos de conteúdo, que são servidos diretamente do seu servidor de origem. Em vez disso, MediaTailor busca os manifestos de sua origem e os usa para montar uma janela de manifesto deslizante ativa que faz referência aos segmentos de conteúdo subjacentes. A montagem do canal acompanha coisas como o número da sequência de mídia, que é necessário para facilitar a reprodução de um ativo para outro. Os fluxos montados lineares são criados com um baixo custo operacional usando conteúdo VOD codificado e empacotado com várias taxas de bits existentes.

Você pode monetizar facilmente os fluxos lineares de montagem de canais inserindo intervalos publicitários em seus programas sem precisar condicionar o conteúdo com os marcadores SCTE-35. Você pode usar a montagem de canais com o serviço de inserção de MediaTailor anúncios ou com qualquer serviço de inserção de anúncios do lado do servidor.

Para começar a montar canais, consulte [the section called “Introdução à montagem MediaTailor de canais”](#).

Para obter informações sobre a integração do Channel Assembly com uma rede de distribuição de conteúdo (CDN), consulte. [Crie canais MediaTailor lineares com montagem de canais e CDN](#)

Tópicos

- [Trabalhando com locais de origem](#)
- [Como trabalhar com canais](#)
- [Adicionar um programa à programação de um canal](#)
- [Insira anúncios e intervalos de anúncios personalizados em um stream do canal](#)
- [Mudando o horário da reprodução de um canal](#)

Trabalhando com locais de origem

Um local de origem representa o servidor de origem em que seu conteúdo de origem está armazenado. Um local de origem pode ser o Amazon S3, um servidor web padrão, uma rede de entrega de conteúdo (CDN), como a Amazon CloudFront, ou uma origem de embalagem, como.

AWS Elemental MediaPackage MediaTailor recupera seus manifestos de conteúdo do local de origem e os usa para montar o fluxo linear do seu canal.

Este tópico explica como usar o AWS Elemental MediaTailor console para criar e excluir locais de origem e como trabalhar com fontes de VOD.

Tópicos

- [Criação de um local de origem](#)
- [Configurando a autenticação para seu local de origem](#)
- [Trabalhando com fontes de VOD](#)
- [Trabalhando com fontes ao vivo](#)
- [Usando configurações de pacotes](#)
- [Cache de manifestos](#)

Criação de um local de origem

O procedimento a seguir explica como criar um local de origem usando o MediaTailor console. Para obter informações sobre como criar locais de origem usando a MediaTailor API, consulte [CreateSourceLocation](#) Referência da AWS Elemental MediaTailor API.

Para criar um local de origem

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem do canal > Locais de origem.
3. Na barra de navegação, escolha Criar local de origem.
4. Em Configuração do local de origem, insira um nome e a URL base do seu servidor de origem:
 - Nome: um identificador para seu local de origem, como my-origin.
 - URL base: o protocolo e o URL base do servidor de origem em que seu conteúdo está armazenado, como <https://111111111111.cloudfront.net>. O URL deve estar em um formato de URL HTTP padrão, prefixado com <http://> ou <https://>.

Opcionalmente, selecione Usar SigV4 para autenticação do Amazon S3 se seu local de origem for um bucket do Amazon S3 e se você quiser usar o Signature versão AWS 4 para autenticação de acesso ao Amazon S3. Para obter informações avançadas, consulte [Configurando a autenticação para seu local de origem](#).

5.

Em Configuração do Access, configure opcionalmente a autenticação para seu local de origem:

- Tipo de acesso: selecione o tipo de autenticação MediaTailor usado para acessar o conteúdo armazenado na origem do local de origem.
- SigV4 para Amazon S3 - MediaTailor usa o Amazon Signature versão 4 (SigV4) para autorizar a solicitação à sua origem. Para obter mais informações, consulte [the section called “Autenticação de solicitações para o Amazon S3 com SigV4”](#).
- Autenticação do token de acesso do Secrets Manager - MediaTailor usa o Secrets Manager e uma chave gerenciada pelo AWS KMS cliente criada, de propriedade e gerenciada por você para facilitar a autenticação do token de acesso entre MediaTailor e sua origem. Para obter informações sobre como configurar a autenticação do token de acesso do Secrets Manager, consulte [the section called “Trabalhando com AWS Secrets Manager autenticação de token de acesso”](#).
- Nome do cabeçalho - especifique um nome de cabeçalho HTTP. MediaTailor usa o cabeçalho HTTP para enviar o token de acesso à sua origem em solicitações de manifesto de conteúdo. Você pode usar qualquer nome de cabeçalho, desde que ele não comece com x-amz- ou x-amzn-. Se você estiver integrando com a [autorização do MediaPackage CDN](#), o valor do cabeçalho deve ser X-MediaPackage-CDNIdentifier
- Chave de sequência de caracteres secreta - A SecretString chave que você especificou no segredo do Secrets Manager. Por exemplo, se você SecretString contém um par de chave e valor, como: {"MyHeaderName": "11111111-2222-3333-4444-111122223333"}, então MyHeaderName é a SecretString chave inserida nesse campo.
- ARN secreto - O ARN do segredo que contém seu token de acesso. Para obter um guia passo a passo, consulte. [Etapa 2: criar um AWS Secrets Manager secret](#)

6. Em Configuração do servidor de entrega por segmentos, configure opcionalmente um servidor para entregar seus segmentos de conteúdo:

- Use um servidor de entrega de segmentos padrão: insira a URL base do servidor que é usada para entregar seus segmentos de conteúdo, como uma CDN. Configure o nome de host do segmento padrão se quiser usar um servidor diferente do servidor de localização de origem para servir os segmentos de conteúdo. Por exemplo, você pode restringir o acesso dos jogadores aos manifestos de origem usando uma configuração de CDN diferente para o URL HTTP básico (o que MediaTailor usa para acessar os manifestos) e o URL base do segmento

padrão (o que os jogadores usam para acessar os segmentos de conteúdo). Se você não inserir um valor, o MediaTailor padrão é o servidor de localização de origem para entrega do segmento.

- Use servidores de entrega por segmentos nomeados: Se você configurou um servidor de entrega por segmento padrão, também pode configurar servidores de entrega por segmentos adicionais. Cada um deve ter um nome exclusivo e uma URL base. O URL base pode ser um URL HTTP completo ou pode ser um caminho relativo, como `/some/path/`. Os nomes são usados para identificar qual servidor deve ser usado ao MediaTailor receber uma solicitação de segmentos de conteúdo. Se a solicitação contiver o cabeçalho `X-MediaTailor-SegmentDeliveryConfigurationName` e o valor do cabeçalho corresponder a um nome, o URL base correspondente será usado para veicular o conteúdo. Se o cabeçalho não estiver incluído na solicitação ou se não corresponder a nenhum nome, o servidor de entrega de segmentos padrão será usado.

7. Escolha Criar local de origem.

8. Para adicionar mais locais de origem, repita as etapas 2 a 6.

Configurando a autenticação para seu local de origem

Use a configuração de acesso para configurar a autenticação para seu local de origem. Quando a configuração de acesso está ativada, MediaTailor só recupera os manifestos de origem de sua origem se a solicitação for autorizada entre MediaTailor e sua origem. A configuração de acesso está desativada por padrão.

MediaTailor suporta os seguintes tipos de autenticação:

- SigV4 para autenticação do Amazon S3
- AWS Secrets Manager token de acesso
- SigV4 para autenticação da MediaPackage versão 2 (v2)

Este capítulo explica como usar o SigV4 para Amazon S3 MediaPackage , v2 AWS Secrets Manager e tokens de acesso para autenticação de localização de origem.

Para obter mais informações, selecione o tópico aplicável.

Tópicos

- [Autenticação de solicitações para o Amazon S3 com SigV4](#)

- [Trabalhando com o SigV4 para MediaPackage a versão 2](#)
- [Trabalhando com AWS Secrets Manager autenticação de token de acesso](#)

Autenticação de solicitações para o Amazon S3 com SigV4

O Signature Version 4 (SigV4) para Amazon S3 é um protocolo de assinatura usado para autenticar solicitações para o Amazon S3 via HTTPS. Quando você usa o SigV4 para o Amazon S3 MediaTailor, inclui um cabeçalho de autorização assinado na solicitação HTTPS para o bucket do Amazon S3 usado como sua origem. Se o cabeçalho de autorização assinado for válido, sua origem atenderá à solicitação. Se não for válido, a solicitação falhará.

Para obter informações gerais sobre o SigV4 for AWS Key Management Service, consulte o tópico [Solicitações de autenticação \(AWS assinatura versão 4\)](#) na referência da API do Amazon S3.

Note

MediaTailor sempre assina solicitações para essas origens com o SigV4.

Requisitos

Se você ativar o SigV4 para autenticação do Amazon S3 em seu local de origem, deverá atender aos seguintes requisitos:

- Você deve permitir MediaTailor o acesso ao seu bucket do Amazon S3 concedendo acesso principal a `mediatailor.amazonaws.com` no IAM. Para obter informações sobre como configurar o acesso no IAM, consulte [Gerenciamento de acesso](#) no Guia do AWS Identity and Access Management usuário.
- O responsável pelo serviço `mediatailor.amazonaws.com` deve ter permissões para ler todas as playlists multivariantes referenciadas pelas configurações do pacote fonte do VOD.
- O chamador da API deve ter permissões `s3: GetObject` IAM para ler todas as playlists multivariantes referenciadas pelas configurações do pacote fonte de VOD. MediaTailor
- Seu URL base de localização de MediaTailor origem deve seguir o formato de URL de solicitação de estilo hospedado virtual do Amazon S3. Por exemplo, `https://bucket-name.s3.Region.amazonaws.com/key-name`. [Para obter informações sobre o acesso em estilo virtual hospedado no Amazon S3, consulte Solicitações virtuais. Hosted-Style](#)

Trabalhando com o SigV4 para MediaPackage a versão 2

O Signature Version 4 (SigV4) para MediaPackage v2 é um protocolo de assinatura usado para autenticar solicitações para MediaPackage a v2 via HTTP. Quando você usa o SigV4 para MediaPackage v2, MediaTailor inclui um cabeçalho de autorização assinado na solicitação HTTP para o endpoint MediaPackage v2 usado como sua origem. Se o cabeçalho de autorização assinado for válido, sua origem atenderá à solicitação. Se não for válido, a solicitação falhará.

Para obter informações gerais sobre o SigV4 para MediaPackage v2, consulte o tópico [Solicitações de autenticação \(AWS assinatura versão 4\)](#) na referência da MediaPackage API v2.

Requisitos

Se você ativar o SigV4 para autenticação MediaPackage v2 em seu local de origem, deverá atender aos seguintes requisitos:

- Você deve permitir MediaTailor o acesso ao seu endpoint MediaPackage v2 concedendo acesso principal à `mediatailor.amazonaws.com` em uma Política de Acesso Origin no endpoint.
- Seu URL base de localização de MediaTailor origem deve ser um endpoint MediaPackage v2.
- O chamador da API deve ter permissões `mediapackagev2:GetObject` IAM para ler todas as playlists multivariantes referenciadas pelas configurações do pacote de origem. MediaTailor

Trabalhando com AWS Secrets Manager autenticação de token de acesso

MediaTailor suporta a autenticação de token de acesso do Secrets Manager. Com a autenticação do token de AWS Secrets Manager acesso, MediaTailor usa uma chave AWS Key Management Service (AWS KMS) gerenciada pelo cliente e um AWS Secrets Manager segredo que você cria, possui e gerencia para autenticar solicitações em sua origem.

Nesta seção, explicamos como a autenticação do token de acesso do Secrets Manager funciona e fornecemos informações passo a passo sobre como configurar a autenticação do token de acesso do Secrets Manager. Você pode trabalhar com a autenticação do token de acesso do Secrets Manager no Console de gerenciamento da AWS ou programaticamente com AWS APIs.

Tópicos

- [Configurar AWS Secrets Manager autenticação de token de acesso](#)
- [Integração com MediaPackage endpoints que usam autorização CDN](#)

- [Como funciona a autenticação do token de acesso do MediaTailor Secrets Manager](#)

Configurar AWS Secrets Manager autenticação de token de acesso

Quando quiser usar a autenticação por token de AWS Secrets Manager acesso, execute as seguintes etapas:

1. Você [cria uma chave gerenciada pelo AWS Key Management Service cliente](#).
2. Você [cria um AWS Secrets Manager segredo](#). O segredo contém seu token de acesso, que é armazenado no Secrets Manager como um valor secreto criptografado. MediaTailor usa a chave gerenciada pelo AWS KMS cliente para descriptografar o valor secreto.
3. Você configura um local AWS Elemental MediaTailor de origem para usar a autenticação do token de acesso do Secrets Manager.

A seção a seguir fornece orientação passo a passo sobre como configurar a autenticação por token de AWS Secrets Manager acesso.

Tópicos

- [Etapa 1: criar um AWS KMS chave simétrica gerenciada pelo cliente](#)
- [Etapa 2: criar um AWS Secrets Manager secret](#)
- [Etapa 3: configurar um local de MediaTailor origem com autenticação de token de acesso](#)

Etapa 1: criar um AWS KMS chave simétrica gerenciada pelo cliente

Você usa AWS Secrets Manager para armazenar seu token de acesso na forma de um segredo `SecretString` armazenado. O `SecretString` é criptografado por meio do uso de uma chave AWS KMS simétrica gerenciada pelo cliente que você cria, possui e gerencia. MediaTailor usa a chave simétrica gerenciada pelo cliente para facilitar o acesso ao segredo com uma concessão e para criptografar e descriptografar o valor secreto.

As chaves gerenciadas pelo cliente permitem que você execute tarefas como as seguintes:

- Estabelecer e manter as políticas de chave
- Estabelecer e manter subsídios e IAM policies
- Habilitar e desabilitar políticas de chaves
- Material de chave criptográfica rotativa

- Adicionar tags da

Para obter informações sobre como o Secrets Manager usa AWS KMS para proteger segredos, consulte o tópico [Como AWS Secrets Manager usar AWS KMS](#) no Guia do AWS Key Management Service Desenvolvedor.

Para obter mais informações sobre chaves gerenciadas pelo cliente, consulte [Chaves gerenciadas pelo cliente](#) no Guia do desenvolvedor do AWS Key Management Service .

Note

AWS KMS cobranças se aplicam ao uso de uma chave gerenciada pelo cliente. Para obter mais informações sobre preços, consulte a página de [AWS Key Management Service preços](#).

Você pode criar uma chave AWS KMS simétrica gerenciada pelo cliente usando Console de gerenciamento da AWS ou programaticamente com as APIs. AWS KMS

Para criar uma chave simétrica gerenciada pelo cliente

Siga as etapas para [criar uma chave simétrica gerenciada pelo cliente](#) no Guia do AWS Key Management Service desenvolvedor.

Anote o nome de recurso principal da Amazon (ARN); você precisará dele. [Etapa 2: criar um AWS Secrets Manager secret](#)

Contexto de criptografia

Um contexto de criptografia é um conjunto opcional de pares chave-valor que pode conter informações contextuais adicionais sobre os dados.

O Secrets Manager inclui um [contexto de criptografia](#) ao criptografar e descriptografar o. SecretString O contexto de criptografia inclui o ARN secreto, que limita a criptografia a esse segredo específico. Como medida adicional de segurança, MediaTailor cria uma AWS KMS concessão em seu nome. MediaTailor aplica uma [GrantConstraints](#) operação que só nos permite descriptografar o ARN SecretString associado ao secreto contido no contexto de criptografia do Secrets Manager.

Para obter informações sobre como o Secrets Manager usa o contexto de [criptografia](#), consulte o [tópico Contexto](#) de criptografia no Guia do AWS Key Management Service desenvolvedor.

Definindo a política principal

As políticas de chaves controlam o acesso à chave gerenciada pelo cliente. Cada chave gerenciada pelo cliente deve ter exatamente uma política de chave, que contém declarações que determinam quem pode usar a chave e como pode usá-la. Ao criar sua chave gerenciada pelo cliente, você pode usar a política de chaves padrão. Para obter mais informações, consulte [Autenticação e controle de acesso para o AWS KMS](#) no Guia do desenvolvedor do AWS Key Management Service .

Para usar sua chave gerenciada pelo cliente com seus recursos de localização de MediaTailor origem, você deve dar permissão ao diretor do IAM que chama [CreateSourceLocation](#) ou [UpdateSourceLocation](#) usa as seguintes operações de API:

- `kms:CreateGrant`— Adiciona uma concessão a uma chave gerenciada pelo cliente. MediaTailor cria uma concessão em sua chave gerenciada pelo cliente que permite que ela use a chave para criar ou atualizar um local de origem configurado com autenticação de token de acesso. Para obter mais informações sobre como usar o [Grants in AWS KMS](#), consulte o Guia do AWS Key Management Service desenvolvedor.

Isso permite MediaTailor fazer o seguinte:

- Ligue `Decrypt` para que ele possa recuperar com sucesso seu segredo do Secrets Manager ao ligar [GetSecretValue](#).
- Ligue `RetireGrant` para retirar a concessão quando o local de origem for excluído ou quando o acesso ao segredo for revogado.

Veja a seguir um exemplo de declaração de política que você pode adicionar para 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"
    }
  }
}
```

```
}
```

Para obter mais informações sobre como especificar permissões em uma política e solucionar problemas de acesso por chave, consulte [Concessões AWS KMS no Guia do AWS Key Management Service](#) desenvolvedor.

Etapa 2: criar um AWS Secrets Manager secret

Use o Secrets Manager para armazenar seu token de acesso na forma de um `SecretString` que é criptografado por uma chave gerenciada pelo AWS KMS cliente. MediaTailor usa a chave para descriptografar o `SecretString`. Para obter informações sobre como o Secrets Manager usa AWS KMS para proteger segredos, consulte o tópico [Como AWS Secrets Manager usar AWS KMS](#) no Guia do AWS Key Management Service Desenvolvedor.

Se você usa AWS Elemental MediaPackage como origem do local de origem e gostaria de usar a autenticação por token de acesso do MediaTailor Secrets Manager, siga o procedimento [the section called "Integração com MediaPackage endpoints que usam autorização CDN"](#).

Você pode criar um segredo do Secrets Manager usando Console de gerenciamento da AWS ou programaticamente com as APIs do Secrets Manager.

Como criar um segredo

Siga as etapas para [Criar e gerenciar segredos AWS Secrets Manager](#) no Guia do AWS Secrets Manager usuário.

Lembre-se das seguintes considerações ao criar seu segredo:

- [KmsKeyId](#) Deve ser o [ARN da chave](#) gerenciada pelo cliente que você criou na Etapa 1.
- Você deve fornecer um [SecretString](#). `SecretString` Deve ser um objeto JSON válido que inclua uma chave e um valor contendo o token de acesso. Por exemplo, `{"MyAccessTokenIdentifier": "112233445566"}`. O valor deve ter entre 8 e 128 caracteres.

Ao configurar seu local de origem com a autenticação do token de acesso, você especifica a `SecretString` chave. MediaTailor usa a chave para pesquisar e recuperar o token de acesso armazenado no `SecretString`.

Anote o ARN secreto e a `SecretString` chave. Você os usará ao configurar seu local de origem para usar a autenticação por token de acesso.

Anexando uma política secreta baseada em recursos

Para permitir o MediaTailor acesso ao valor secreto, você deve anexar uma política baseada em recursos ao segredo. Para obter mais informações, consulte [Anexar uma política de permissões a um segredo do Secrets Manager](#) no Guia AWS Secrets Manager do Usuário.

Veja a seguir um exemplo de declaração de política que você pode adicionar para MediaTailor:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "mediatailor.amazonaws.com"
      },
      "Action": "secretsmanager:GetSecretValue",
      "Resource": "arn:aws:secretsmanager:us-  
east-1:111122223333:secret:secret-name"
    }
  ]
}
```

Etapa 3: configurar um local de MediaTailor origem com autenticação de token de acesso

Você pode configurar a autenticação do token de acesso do Secrets Manager usando Console de gerenciamento da AWS ou programaticamente com as MediaTailor APIs.

Para configurar um local de origem com a autenticação de token de acesso do Secrets Manager

Siga as etapas [Access configuration](#) do Guia do AWS Elemental MediaTailor usuário.

Integração com MediaPackage endpoints que usam autorização CDN

Se você usa AWS Elemental MediaPackage como origem de localização de origem, MediaTailor pode se integrar com MediaPackage endpoints que usam autorização CDN.

Para fazer a integração com um MediaPackage endpoint que usa autorização CDN, use o procedimento a seguir.

Para integrar com MediaPackage

1. Conclua as etapas em [Configurar a autorização de CDN](#) no Guia AWS Elemental MediaPackage do usuário, caso ainda não tenha feito isso.
2. Conclua o procedimento em [the section called “Etapa 1: criar um AWS KMS chave simétrica gerenciada pelo cliente”](#).
3. Modifique o segredo que você criou ao configurar a autorização de MediaPackage CDN. Modifique o segredo com os seguintes valores:
 - Atualize o KmsKeyId com o ARN da chave gerenciada pelo cliente que você criou em [the section called “Etapa 1: criar um AWS KMS chave simétrica gerenciada pelo cliente”](#)
 - (Opcional) Para oSecretString, você pode alternar o UUID para um novo valor ou usar o segredo criptografado existente, desde que seja um par de chave e valor em um formato JSON padrão, como. {"MediaPackageCDNIdentifier": "112233445566778899"}
4. Siga as etapas em [the section called “Anexando uma política secreta baseada em recursos”](#).
5. Siga as etapas em [the section called “Etapa 3: configurar um local de MediaTailor origem com autenticação de token de acesso”](#).

Como funciona a autenticação do token de acesso do MediaTailor Secrets Manager

Depois de criar ou atualizar um local de origem para usar a autenticação do token de acesso, MediaTailor inclua o token de acesso em um cabeçalho HTTP ao solicitar manifestos de conteúdo de origem de sua origem.

Aqui está uma visão geral de como MediaTailor usa a autenticação de token de acesso do Secrets Manager para autenticação de origem do local de origem:

1. Quando você cria ou atualiza um local de MediaTailor origem que usa autenticação de token de acesso, MediaTailor envia uma [DescribeSecret](#) solicitação ao Secrets Manager para determinar a AWS KMS chave associada ao segredo. Você inclui o ARN secreto na configuração de acesso à localização de origem.
2. MediaTailor cria uma [concessão](#) para a chave gerenciada pelo cliente, para que MediaTailor possa usar a chave para acessar e descriptografar o token de acesso armazenado no.

SecretString O nome do subsídio será MediaTailor-SourceLocation-*your Conta da AWS ID-source location name*.

Você pode revogar o acesso à concessão ou remover MediaTailor o acesso à chave gerenciada pelo cliente a qualquer momento. Para obter mais informações, consulte [RevokeGrant](#) na Referência de APIs do AWS Key Management Service .

3. Quando uma fonte de VOD é criada, atualizada ou usada em um programa, MediaTailor faz solicitações HTTP aos locais de origem para recuperar os manifestos de conteúdo de origem associados às fontes de VOD no local de origem. Se a fonte de VOD estiver associada a um local de origem que tenha um token de acesso configurado, as solicitações incluirão o token de acesso como um valor de cabeçalho HTTP.

Trabalhando com fontes de VOD

Uma fonte de VOD representa um único conteúdo, como um vídeo ou um episódio de um podcast, que você adiciona ao local de origem. Você adiciona uma ou mais fontes de VOD ao local de origem e associa cada fonte de VOD a um programa depois de criar seu canal.

Cada fonte de VOD deve ter pelo menos uma configuração de pacote. Uma configuração de pacote especifica um formato de pacote, localização do manifesto e grupo de origem para sua fonte de VOD. Ao criar seu canal, você usa os grupos de origem da configuração do pacote para criar as saídas correspondentes em seu canal. Por exemplo, se sua fonte estiver empacotada em dois formatos diferentes — HLS e DASH —, você criará duas configurações de pacote, uma para DASH e outra para HLS. Em seguida, você criaria duas saídas de canal, uma para cada configuração de pacote. Cada saída de canal fornece um endpoint que é usado para solicitações de reprodução. Portanto, usando o exemplo anterior, o canal forneceria um endpoint para solicitações de reprodução HLS e um endpoint para solicitações de reprodução DASH.

Se você quiser que os deslocamentos dos marcadores de anúncios em seu manifesto sejam detectados automaticamente, cada marcador de anúncio deve aparecer com o mesmo deslocamento em todas as configurações do pacote e ter uma duração de zero. Para HLS, MediaTailor detectará DATERANGE e marcará EXT-X-CUE-OUT. Para o DASH, o HLS detectará a primeira tag de evento em cada EventStream tag.

No exemplo a seguir, uma oportunidade de pausa publicitária será detectada com um deslocamento de 12.000 ms por causa da DATERANGE tag com uma duração de 0,0. A primeira DATERANGE tag com um deslocamento de 0 ms não será detectada porque tem uma duração de 10,0.


```
#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
```

No exemplo a seguir, uma oportunidade de interrupção publicitária será detectada com um deslocamento de 0 ms porque a EXT-X-CUE-OUT tag tem uma duração de 0 e é seguida imediatamente por uma EXT-X-CUE-IN tag. O segundo EXT-X-CUE-IN parEXT-X-CUE-OUT/não será detectado porque tem uma duração de 10.

```
#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
```

No exemplo a seguir, uma oportunidade de pausa publicitária será detectada com uma diferença de 0 ms porque o primeiro evento EventStream ocorre no período que começa às. PT0.000S O segundo Event no não EventStream será detectado.

```
<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>

```

Adicionando fontes de VOD à sua localização de origem

O procedimento a seguir explica como adicionar fontes de VOD à sua localização de origem e definir configurações de pacotes usando o MediaTailor console. Para obter informações sobre como adicionar fontes de VOD usando a MediaTailor API, consulte [CreateVodSource](#) na Referência da AWS Elemental MediaTailor API.

Important

Antes de adicionar suas fontes de VOD, verifique se elas atendem aos seguintes requisitos:

- Todas as variantes de origem devem ter o mesmo tamanho, conforme determinado pelo manifesto de origem.
- Em uma configuração de pacote, cada fonte deve ter o mesmo número de fluxos secundários.

Devido a esses requisitos, não oferecemos suporte para ABR por título ou automatizado, pois esses métodos de codificação podem produzir diferentes comprimentos de manifestos e fluxos secundários.

Recomendamos que você use um modelo de codificação que inclua um comprimento mínimo de segmento para garantir que suas fontes codificadas atendam a esses requisitos.

Para adicionar fontes de VOD aos seus locais de origem

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem do canal > Locais de origem.
3. No painel Locais de origem, escolha o local de origem que você criou no [Para criar um local de origem](#) procedimento.
4. Escolha Adicionar fonte de VOD.
5. Em Detalhes da fonte de VOD, insira um nome para sua fonte de VOD:
 - Nome: Um identificador para sua fonte de VOD, como my-example-video.
6. Em Package configurations > *source-group-name* insira as informações sobre a configuração do pacote:

 Note

Todas as configurações do pacote de origem devem ter a mesma duração, conforme determinado pelo manifesto da fonte. Além disso, todas as fontes em uma configuração de pacote devem ter o mesmo número de fluxos secundários. Para atender a esses requisitos, recomendamos que você use um modelo de codificação para seus ativos. Recomendamos que você use um modelo de codificação com um comprimento mínimo de segmento de um segundo. MediaTailor não oferece suporte por título ou streaming automático de taxa de bits adaptativa (ABR) porque esses métodos de codificação violam esses requisitos.

- Grupo de origem: insira um nome de grupo de origem que descreva essa configuração de pacote, como HLS-4k. Anote esse nome; você o referenciará ao criar a saída do seu canal. Para obter mais informações, consulte [Usando grupos de origem com as saídas do seu canal](#).

- Tipo: Selecione o formato do pacote para essa configuração. MediaTailor suporta HLS e DASH.
- Caminho relativo: o caminho relativo do URL HTTP base do local de origem até o manifesto. Por exemplo, /my/path/index.m3u8.

 Note

MediaTailor importa automaticamente todas as legendas ocultas e streams secundários contidos em uma playlist multivariante principal. Você não precisa criar configurações de pacotes separadas para cada uma das suas rendições de origem (DASH) ou fluxos de variantes (HLS).

Para obter mais informações sobre configurações de pacotes, consulte [Usando configurações de pacotes](#).

7. Escolha Adicionar fonte de VOD.

Se você quiser adicionar mais fontes de VOD, repita as etapas de 4 a 7 no procedimento.

Trabalhando com fontes ao vivo

Uma fonte ao vivo representa uma única transmissão ao vivo, como um jogo de futebol ao vivo ou um noticiário, que você adiciona ao local de origem. Depois de criar seu canal, você adiciona uma ou mais fontes ao vivo ao local de origem e associa cada fonte ao vivo a um programa.

MediaTailor suporta esses tipos de montagem de canais lineares:

- Fontes de VOD para um canal que contém conteúdo VOD-to-live
- Fontes ao vivo para um canal que contém conteúdo ao vivo misturado com conteúdo VOD-to-live

Um exemplo de VOD-to-live conteúdo é um canal que reúne uma biblioteca de ativos de VOD em uma transmissão ao vivo. Um exemplo de conteúdo ao vivo misturado com conteúdo é um canal que mostra principalmente VOD-to-live conteúdo VOD, exceto um noticiário noturno ou um evento esportivo ao vivo pré-agendado. Outro exemplo de conteúdo ao vivo misturado com VOD-to-live conteúdo é um canal totalmente ao vivo com origens que variam de acordo com a hora do dia.

Você pode usar fontes ao vivo para configurar um canal regional que mostre principalmente a programação nacional, mas também inclua substituições de programação regional e tenha conteúdo VOD misturado. Para fazer isso, você executa um encoder/packager para o conteúdo nacional e, em seguida, executa codificadores regionais quando essas regiões estão ativas. Em seguida, você cria canais regionais de montagem de canais, cada um com seus próprios horários. Dessa forma, os espectadores podem alternar entre si conforme necessário. Essa configuração ajuda você a minimizar encoding/packaging os custos.

Cada fonte ativa deve ter pelo menos uma configuração de pacote. Uma configuração de pacote especifica um formato de pacote, localização do manifesto e grupo de origem para sua fonte ativa. Ao criar seu canal, você usa os grupos de origem da configuração do pacote para criar as saídas correspondentes em seu canal. Por exemplo, se sua fonte estiver empacotada em dois formatos diferentes — HLS e DASH —, você criará duas configurações de pacote, uma para DASH e outra para HLS. Em seguida, você criaria duas saídas de canal, uma para cada configuração de pacote. Cada saída de canal fornece um endpoint que é usado para solicitações de reprodução. Neste exemplo, o canal fornece um endpoint para solicitações de reprodução HLS e um endpoint para solicitações de reprodução DASH.

Requisitos gerais para o uso de fontes ativas

Quando você usa fontes ativas, seu conteúdo deve estar alinhado com os seguintes requisitos gerais:

- Fontes ativas do HLS - Você deve fornecer #EXT-X-PROGRAM-DATE-TIME tags para o primeiro segmento na janela do manifesto e em cada descontinuidade.
- HLS - Você deve configurar os marcadores de anúncios como. DATERANGE
- Janela de manifesto de origem - recomendamos usar uma janela de manifesto com uma duração pelo menos tão longa quanto a janela de manifesto em seu MediaTailor canal Channel Assembly. Como prática recomendada, considere usar uma duração de janela de manifesto de 30 segundos ou mais do que a janela de manifesto no canal Channel Assembly.
- Faça com que a duração alvo corresponda à duração das fontes existentes.
- Faça com que o número de playlists infantis corresponda ao das fontes existentes.

Configurações

Se você usa outros serviços de mídia AWS Elemental como parte do seu fluxo de trabalho de fontes ativas, recomendamos seguir as melhores práticas ao definir sua MediaPackage configuração. A

tabela a seguir descreve como definir MediaPackage as configurações com base no padrão de streaming que você usa.

MediaPackage configuração para fontes ao vivo

Standard	Configuração	Valor	necessidade	Observações
HLS	Endpoint type	Apple HLS	Obrigatório, a menos que use o CMAF	Para corresponder às tarefas do HLS ts AWS Elemental MediaConvert
HLS	Endpoint type	CMAF	Obrigatório, a menos que use o Apple HLS	Para corresponder às tarefas do HLS mp4 AWS Elemental MediaConvert
HLS	ProgramDateTimeIntervalSeconds	1	Obrigatório	Você deve especificar #EXT-X-PROGRAM-DATE-TIME em cada segmento para evitar problemas de reprodução quando houver descontinuidades.
HLS	PlaylistWindowSeconds	30 segundos a mais do que a janela do manifesto de montagem do canal	Obrigatório	

Standard	Configuração	Valor	necessidade	Observações
HLS	AdMarkers	DATERANGE	Obrigatório ao passar por marcadores de anúncios	
HLS	IncludeIframeOnlyStream	Desabilitado	Recomendado	
DASH	ManifestLayout	FULL	Recomendado	
DASH	SegmentTemplateFormat	NUMBER_WITH_TIMELINE ou TIME_WITH_TIMELINE	Recomendado	Não há suporte ao NUMBER_WITH_DURATION .
DASH	ManifestWindowSeconds	30 segundos a mais do que a janela do manifesto de montagem do canal	Obrigatório	
DASH	PeriodTriggers	ADS	Obrigatório ao passar por marcadores de anúncios	

Adicionar fontes ativas ao seu local de origem

O procedimento a seguir explica como usar o MediaTailor console para adicionar fontes ativas ao seu local de origem e definir as configurações do pacote. Para obter informações sobre como

adicionar fontes ativas usando a MediaTailor API, consulte [CreateLiveSource](#) a Referência da AWS Elemental MediaTailor API.

 Important

Antes de adicionar suas fontes ativas, certifique-se de que, dentro de uma configuração de pacote, cada fonte tenha o mesmo número de streams secundários.

Para adicionar fontes ativas aos seus locais de origem

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem do canal > Locais de origem.
3. No painel Locais de origem, escolha o local de origem que você criou no [Para criar um local de origem](#) procedimento.
4. Na guia Fontes ativas, escolha Adicionar fonte ativa.
5. Em Detalhes da fonte ativa, insira um nome para sua fonte ativa:
 - Nome: um identificador para sua fonte ao vivo, como my-example-video.
6. Em Package configurations > *source-group-name* insira as informações sobre a configuração do pacote:

 Note

Em uma configuração de pacote, todas as fontes de VOD e fontes ativas devem ter o mesmo número de fluxos secundários. Recomendamos que você configure seus fluxos de origem da mesma forma.

- Grupo de origem: insira um nome de grupo de origem que descreva essa configuração de pacote, como HLS-4k. Anote esse nome; você o referenciará ao criar a saída do seu canal. Para obter mais informações, consulte [Usando grupos de origem com as saídas do seu canal](#).
- Tipo: Selecione o formato do pacote para essa configuração. MediaTailor suporta HLS e DASH.
- Caminho relativo: o caminho relativo do URL HTTP base do local de origem até o manifesto. Por exemplo, /my/path/index.m3u8.

 Note

MediaTailor importa automaticamente todas as legendas ocultas e streams secundários contidos em uma playlist multivariante principal. Você não precisa criar configurações de pacotes separadas para cada uma das suas rendições de origem (DASH) ou fluxos de variantes (HLS).

Para obter mais informações sobre configurações de pacotes, consulte [Usando configurações de pacotes](#).

7. Escolha Adicionar fonte ao vivo.

Se você quiser adicionar mais fontes ativas, repita as etapas de 4 a 6 no procedimento.

Usando configurações de pacotes

Uma configuração de pacote é uma representação da fonte que contém as várias características de embalagem necessárias para reprodução em dispositivos diferentes. Por exemplo, você pode ter uma fonte que tenha três formatos de pacote: HLS com DRM, DASH com endereçamento da linha do tempo do segmento e HLS com segmentos CMAF.

A montagem do canal não reempacota suas fontes. Se quiser incluir vários formatos empacotados para uma determinada fonte, você deve disponibilizar cada formato empacotado no local de origem e especificar o caminho para cada formato empacotado.

Cada objeto de configuração do pacote deve incluir o seguinte:

- Caminho relativo - O caminho completo para o formato empacotado da fonte, em relação ao local de origem. Por exemplo, `/my/path/index.m3u8`.
- Grupo de origem - O nome do grupo de origem usado para associar as configurações do pacote à saída de um canal.
- Tipo - HLS ou DASH.

Depois de criar um canal, você também deve declarar cada grupo de origem que deseja usar para a saída do canal.

Cache de manifestos

MediaTailor armazena periodicamente e de forma oportunista as playlists de origem para melhorar o desempenho e a confiabilidade da montagem de canais. Às vezes, a versão em cache fica obsoleta em comparação com a versão original no local de origem. Para MediaTailor forçar a atualização da versão em cache da fonte, chame. [UpdateVodSource](#) Por exemplo, use essa chamada quando os caminhos incorporados mudarem em sua fonte. Certifique-se de sempre manter uma versão atualizada da fonte disponível em seu local de origem, mesmo se você receber poucas solicitações de MediaTailor.

Como trabalhar com canais

Um canal reúne seus manifestos de origem em um fluxo linear. Cada canal contém uma ou mais saídas que correspondem às configurações do seu pacote.

Primeiro, você cria um canal, depois adiciona suas fontes de VOD e fontes ao vivo à programação do canal criando programas. Cada programa está associado a uma fonte VOD ou a uma fonte ativa.

Tópicos

- [Crie um canal usando o MediaTailor console](#)
- [Usando grupos de origem com as saídas do seu canal](#)
- [Excluir um canal usando o MediaTailor console](#)

Crie um canal usando o MediaTailor console

O procedimento a seguir descreve como criar um canal usando o MediaTailor console.

Para criar um canal

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. Na barra de navegação, escolha Criar canal.
4. Em Detalhes do canal, insira detalhes sobre seu canal:
 - Nome: insira um nome para seu canal.

- **Nível:** o nível determina quais recursos o canal suporta e quanto custa administrar o canal. Para obter mais informações sobre preços, consulte a [página de preços do Channel Assembly](#). MediaTailor suporta os seguintes níveis:

- **Básico** - O nível Básico suporta os modos de reprodução Linear e Loop e não suporta fontes ao vivo.
- **Padrão** - O nível Padrão suporta fontes ao vivo e requer o modo de reprodução linear.

Ao selecionar Padrão nos detalhes do canal, você pode definir os públicos em Detalhes do público. Esses públicos serão usados para ProgramRules quando você criar o AudienceMedia para seu programa padrão.

- Escolha Adicionar.
 - Insira o nome do público na caixa de texto. Ele deve ter entre 1 e 32 caracteres alfanuméricos.
 - Escolha Confirmar.
 - Escolha Próximo.
- **Modo de reprodução:** o modo de reprodução define o comportamento de reprodução do canal. MediaTailor suporta os seguintes modos de reprodução:
 - **Loop** - Os programas da programação são reproduzidos back-to-back em um loop infinito. Depois que o último programa é reproduzido em um cronograma, a reprodução volta ao primeiro programa. A reprodução continua em loop até você parar o canal.
 - **Linear** - Cada programa na programação é reproduzido uma vez, back-to-back.

5.

Para Filler slate, selecione o nome do local de origem que faz referência ao local do slate e o nome da fonte VOD a ser usado como slate. MediaTailor usa a lista para preencher lacunas entre os programas na programação. Se a duração da lista for menor que a duração da lacuna entre os programas, repetirá a MediaTailor lista. Você deve configurar o campo de preenchimento se seu canal usar o modo de reprodução linear. MediaTailor não suporta filler slate para o modo de reprodução em loop.

6. Escolha Próximo.

7. Especifique os detalhes do público nas regras do programa.

8. Ao selecionar Padrão nos detalhes do canal, você pode definir os públicos em Detalhes do público. Esses públicos serão usados como ProgramRules quando você criar o AudienceMedia para seu programa padrão:

- Selecione Adicionar e, em seguida, adicione um público na caixa de texto e selecione Confirmar.

 Note

Insira um nome que não tenha mais de _ 32 caracteres alfanuméricos.

- Tipo de saída: selecione o formato de streaming para o canal. DASH e HLS são suportados.
 - Grupo de origem: insira o nome do grupo de origem que você criou na configuração do pacote, conforme descrito em [Adicionando fontes de VOD à sua localização de origem](#).
9. Escolha Próximo.
10. Em Configurações do manifesto, insira informações adicionais sobre suas configurações de manifesto:
- Janela de manifesto (s): a janela de tempo, em segundos, contida em cada manifesto. O valor mínimo é 30 segundos e o valor máximo é 3600 segundos.
 - Tipo de marcação de anúncio (somente saídas HLS): o tipo de tags de anúncio que aparecem nos intervalos publicitários do programa VOD. Selecione Daterange para MediaTailor inserir quebras de anúncios em programas de VOD com EXT-X-DATERANGE tags. Selecione Scte35 Enhanced para MediaTailor inserir quebras de anúncios em programas de VOD usando EXT-X-CUE-OUT EXT-X-CUE-IN tags. Para obter mais informações sobre esses tipos de tag, consulte [SCTE-35 mensagens para pausas publicitárias](#). Para fluxos de trabalho ativos, MediaTailor sempre passa por DATERANGE tags e não passa por nenhuma tag Scte35 aprimorada, independentemente do tipo de marcação de anúncio selecionado.
11. Se você quiser configurar saídas de vários canais, em Saídas, escolha Adicionar. Em seguida, configure os detalhes da saída concluindo as etapas 6 e 7 neste procedimento.
12. Escolha Próximo.
13. Em Política do canal, escolha as configurações da política de IAM do seu canal:
- Não anexe a política do canal: restrinja a reprodução somente para aqueles que têm acesso às credenciais dessa conta.
 - Anexe uma política personalizada: defina sua própria política e restrinja o acesso a quantas ou quantas quiser.
 - Anexe uma política pública: aceite todas as solicitações recebidas do cliente à saída de um canal. Você deve usar essa opção se quiser usar a inserção de MediaTailor anúncios.

14. Escolha Próximo.
15. Revise suas configurações no painel Revisar e criar.
16. Escolha Create channel (Criar canal).

 Note

Os canais são criados em um estado parado. Seu canal não estará ativo até que você o inicie com o MediaTailor console ou a MediaTailor StartChannel API.

Usando grupos de origem com as saídas do seu canal

Um grupo de origem associa uma configuração de pacote a uma saída em um canal. Ao criar a configuração do pacote na fonte, você identifica o nome do grupo de origem. Em seguida, ao criar a saída no canal, você insere o mesmo nome para associar a saída à configuração do pacote. As fontes de VOD e as fontes ativas adicionadas a um programa em um canal devem pertencer ao grupo de origem identificado na saída.

Por exemplo:

- As fontes de VOD 1 e 2 têm três configurações de pacotes que têm os grupos de origem: HLS, DASH e HLS-4k.
- A fonte VOD 3 tem duas configurações de pacotes com grupos de origem HLS e DASH.

Se o canal A tiver duas saídas com grupos de origem HLS e DASH, a saída do canal poderá usar todas as três fontes de VOD. Isso porque todas as fontes VOD 1, 2 e 3 têm configurações de pacotes com rótulos de grupos de origem HLS e DASH.

Se o canal B tiver duas saídas com grupos de origem HLS e HLS-4k, ele poderá usar as fontes VOD 1 e 2, mas não 3. Isso ocorre porque as fontes de VOD 1 e 2 têm configurações de pacotes com rótulos de grupos de origem HLS e HLS-4k.

Se o canal C tiver uma única saída com o grupo de origem DASH, ele poderá usar todas as três fontes de VOD. Todas as três fontes de VOD têm configurações de pacotes com o grupo de origem DASH.

Excluir um canal usando o MediaTailor console

Para excluir seu canal, conclua o procedimento a seguir.

Para excluir seu canal

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. Selecione o canal que você deseja excluir.
4. Se seu canal estiver em execução, no menu suspenso Ações, escolha Parar. Você deve interromper seu canal antes de excluí-lo.
5. Quando seu canal estiver parado, no menu suspenso Ações, escolha Excluir.

Adicionar um programa à programação de um canal

Cada programa contém uma fonte de VOD ou uma fonte ativa que faz parte de um local de origem em sua conta. Você adiciona seus programas à programação do seu canal para controlar a ordem em que são reproduzidos na transmissão do seu canal.

Um programa que contém uma fonte de VOD pode ser configurado com um ou mais intervalos publicitários. Cada intervalo de anúncio contém uma lista, que é uma fonte de VOD de um local de origem. Para criar o intervalo publicitário, você adiciona a lista com um deslocamento em milissegundos ao programa.

Tópicos

- [Criação de um programa dentro da programação de um canal usando o MediaTailor console](#)
- [Definindo coortes de público e conteúdo alternativo com as Regras do Programa](#)
- [Gerando manifestos específicos para o público](#)

Criação de um programa dentro da programação de um canal usando o MediaTailor console

O procedimento a seguir descreve como criar um programa dentro da programação do seu canal usando o MediaTailor console. Também descreve como configurar intervalos publicitários, que são opcionais. Para obter informações sobre como criar programas usando a MediaTailor API, consulte [CreatePrograma](#) Referência da AWS Elemental MediaTailor API.

Para adicionar um programa

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. No painel Canais, escolha o canal que você criou no [Para criar um canal](#) procedimento.
4. Nos detalhes do programa, insira detalhes sobre seu programa:
 - Nome: Esse é o nome do programa que você adiciona ao seu canal.
 - Tipo de fonte: determina o tipo de vídeo de origem que o programa reproduz. Essa opção está disponível somente para canais padrão.
 - VOD - O programa reproduz uma video-on-demand fonte, como um episódio de TV pré-gravado.
 - Ao vivo - O programa reproduz uma fonte ao vivo, como uma transmissão de notícias ao vivo.
 - Nome do local de origem: o local de MediaTailor origem associado ao programa.
 - Se você escolher Selecionar um local de origem existente, escolha um nome de local de origem no menu Selecionar um local de origem. Como alternativa, pesquise o local de origem pelo nome. Isso é útil se você tiver um grande número de locais de origem.
 - Se você escolher Inserir o nome do local de origem, pesquise o local de origem pelo nome.
 - Nome da fonte VOD: O nome da fonte VOD MediaTailor associada ao programa:
 - Se você escolher Selecionar uma fonte de VOD existente, selecione um nome de fonte de VOD na lista de fontes de VOD associadas à sua conta. Como alternativa, pesquise sua fonte de VOD pelo nome. Isso é útil se você tiver um grande número de fontes de VOD.
 - Se você escolher Pesquisar por nome, pesquise sua fonte ativa pelo nome.
 - Nome da fonte ativa: o nome da fonte ativa a ser associada ao programa. Essa opção só estará disponível se você tiver selecionado Ao vivo como o tipo de fonte.
 - Se você escolher Selecionar um local de origem existente, escolha um nome de local de origem no menu Selecionar um local de origem. Como alternativa, pesquise o local de origem pelo nome. Isso é útil se você tiver um grande número de locais de origem.
 - Se você escolher Inserir o nome do local de origem, pesquise o local de origem pelo nome.
 - Nome da fonte VOD: O nome da fonte VOD MediaTailor associada ao programa:
 - Se você escolher Selecionar uma fonte ativa existente, selecione um nome de fonte ativa na lista de fontes ativas associadas à sua conta. Como alternativa, você pode

pesquisar sua fonte ao vivo pelo nome. Isso é útil se você tiver um grande número de fontes ativas.

- Se você escolher Pesquisar por nome, pesquise sua fonte ativa pelo nome.

5. Selecione Avançar para ir até a guia Configuração da Programação.

6. Em Configuração de reprodução, defina quando um programa é reproduzido na programação do seu canal:

- Duração em milissegundos: define a duração do programa em milissegundos. Essa opção só está disponível para programas que usam fontes ativas.
- Tipo de transição: define as transições de um programa para outro no cronograma:
 - Relativo: O programa é reproduzido antes ou depois de outro programa na programação. Essa opção só está disponível para programas que usam fontes VOD.
 - Absoluto: o programa é reproduzido em um horário específico do relógio de parede. MediaTailor faz o possível para reproduzir o programa no horário especificado por você. MediaTailor inicia a reprodução do programa em um limite de segmento comum entre o programa ou a lousa anterior. Essa opção está disponível somente para canais configurados para usar o linear [Modo de reprodução: o modo de reprodução define o comportamento de reprodução do canal. MediaTailor suporta os seguintes modos de reprodução:](#).
- Hora de início do programa: para tipos de transição absoluta, a hora do relógio de parede em que o programa está programado para ser reproduzido. Se você estiver adicionando esse programa a um canal linear em execução, deverá inserir uma hora de início que seja 15 minutos ou mais tarde a partir da hora atual.
- Posição relativa: escolha onde inserir o programa na programação, em relação a outro programa. Você pode selecionar Antes do programa ou Depois do programa. Essa configuração não se aplica se esse for o primeiro programa na programação do seu canal.
 - Se você escolher Selecionar um programa existente, selecione o nome do programa em uma lista predefinida dos próximos 100 programas reproduzidos pelo canal no menu Usar programa existente.
 - Se você escolher Pesquisar um programa pelo nome, insira o nome de um programa existente em seu canal.

Se você quiser adicionar intervalos publicitários ao seu programa, vá para a próxima etapa. Os intervalos publicitários só podem ser configurados para programas que usam fontes de

VOD. Para fontes ativas, os intervalos de anúncios nos manifestos do DASH e os intervalos nos manifestos do HLS que usam a EXT-X-DATERANGE tag são transmitidos automaticamente.

7. Selecione Avançar para ir até Adicionar intervalos publicitários.
8. Selecione Adicionar intervalo publicitário. Em Intervalos publicitários, defina as configurações do intervalo publicitário:
 - Nome do local de origem do Slate: escolha Selecionar um local de origem existente e escolha o local de origem em que o Slate está armazenado e que você criou anteriormente nesta tarefa.
 - Nome da fonte de VOD: escolha Selecionar uma fonte de VOD existente e escolha a fonte de VOD que você está usando para o slate que você adicionou anteriormente nesta tarefa. A duração da lista determina a duração do intervalo publicitário.
 - Deslocamento em milissegundos: esse valor determina o horário de início do intervalo publicitário em milissegundos, como um deslocamento em relação ao início do programa. Insira qualquer valor menor que a duração da fonte de VOD e que esteja alinhado com um limite de segmento em todas as faixas da fonte VOD do programa (todas as faixas de áudio, vídeo e legendas ocultas), caso contrário, o intervalo do anúncio será ignorado. Por exemplo, se você inserir 0, isso criará um intervalo publicitário pré-lançamento que será reproduzido antes do início do programa.
 - Número de disponibilidade: [MediaTailor grava em que esse valor é `gravadosplice\_insert.avail\_num`, conforme definido na seção 9.7.3.1. da especificação SCTE-35, Mensagem de sinalização de inserção de programa digital](#). O valor padrão é 0. Os valores devem estar entre 0 e 256, inclusive.
 - Disponibilidade esperada: MediaTailor grava esse valor `emsplice_insert.avails_expected`, conforme definido na seção 9.7.3.1. da especificação SCTE-35. O valor padrão é 0. Os valores devem estar entre 0 e 256, inclusive.
 - ID do evento Splice: MediaTailor grava esse valor `emsplice_insert.splice_event_id`, conforme definido na seção 9.7.3.1. da especificação SCTE-35. O valor padrão é 1.
 - ID exclusivo do programa: MediaTailor grava esse valor `emsplice_insert.unique_program_id`, conforme definido na seção 9.7.3.1. da especificação SCTE-35. O valor padrão é 0. Os valores devem estar entre 0 e 256, inclusive.
9. Para um canal linear padrão, selecione Avançar para ir até Definir mídia alternativa.

Para obter mais informações sobre MediaTailor como usar para criar mídia alternativa, consulte [Criação de mídia alternativa](#).

Para obter informações mais avançadas sobre MediaTailor como personalizar seus intervalos publicitários, consulte [Insira anúncios e intervalos de anúncios personalizados em um stream do canal](#).

10. Selecione Avançar para ir até Revisar e criar.

11. Selecione Adicionar programa.

Para obter informações mais avançadas sobre MediaTailor como personalizar seus intervalos publicitários, consulte [Insira anúncios e intervalos de anúncios personalizados em um stream do canal](#).

12.

 Important

Para canais em loop, se você modificar a lista de programas de um programa programado para os próximos 10 minutos, a edição não ficará aparente até o próximo loop.

Em Detalhes do programa, insira detalhes sobre seu programa:

- Nome: Esse é o nome do programa que você adiciona ao seu canal.
- Tipo de fonte: determina o tipo de fonte que o programa reproduz. Essa opção está disponível somente para canais padrão.
 - VOD - O programa reproduz uma fonte VOD, como um episódio de TV pré-gravado.
 - Ao vivo - O programa reproduz uma fonte ao vivo, como uma transmissão de notícias ao vivo.
- Nome do local de origem: O local de origem a ser associado ao programa.

Se você escolher Selecionar um local de origem existente, selecione um nome de local de origem no menu suspenso Selecionar um local de origem. Como alternativa, você pode pesquisar o local de origem pelo nome. Isso é útil se você tiver um grande número de locais de origem.

Se você escolher Inserir o nome do local de origem, pesquise o local de origem pelo nome.

- Nome da fonte VOD: O nome da fonte VOD a ser associada ao programa.

Se você escolher Selecionar uma fonte de VOD existente, selecione um nome de fonte de VOD na lista de fontes de VOD associadas à sua conta. Como alternativa, você pode

pesquisar sua fonte de VOD pelo nome. Isso é útil se você tiver um grande número de fontes de VOD.

Se você escolher Pesquisar por nome, pesquise sua fonte de VOD por nome.

- Nome da fonte ativa: o nome da fonte ativa a ser associada ao programa. Essa opção só estará disponível se você selecionar Ao vivo como o tipo de fonte.

Se você escolher Selecionar uma fonte ativa existente, selecione um nome de fonte ativa na lista de fontes ativas associadas à sua conta. Como alternativa, você pode pesquisar sua fonte ao vivo pelo nome. Isso é útil se você tiver um grande número de fontes ativas.

Se você escolher Pesquisar por nome, pesquise sua fonte ativa pelo nome.

13. Em Configuração de reprodução, defina quando um programa é reproduzido na programação do seu canal:

- Duração em milissegundos: define a duração do programa em milissegundos. Essa opção só está disponível para programas que usam fontes ativas.
- Tipo de transição: define as transições de um programa para outro no cronograma.
 - Relativo - O programa é reproduzido antes ou depois de outro programa na programação. Essa opção só está disponível para programas que usam fontes VOD.
 - Absoluto - O programa é reproduzido em um horário específico do relógio de parede. MediaTailor faz o possível para reproduzir o programa no horário especificado por você. Iniciamos a reprodução do programa em um limite de segmento comum entre o programa ou a lousa anterior. Essa opção só está disponível para canais configurados para usar [linear playback mode](#) o.

 Note

Esteja ciente do seguinte comportamento para tipos de transição absoluta:

- Se o programa anterior na programação tiver uma duração que se estenda além da hora do relógio de parede, MediaTailor trunca o programa anterior no limite do segmento comum mais próximo da hora do relógio de parede.
- Se houver lacunas entre os programas na programação, MediaTailor joga [filler slate](#). Se a duração da ardósia for menor que a duração da lacuna, faça um MediaTailor loop na ardósia.

- Hora de início do programa - Para tipos de transição absoluta, a hora do relógio de parede em que o programa está programado para ser reproduzido. Se você estiver adicionando esse programa a um canal linear em execução, deverá inserir uma hora de início que seja 15 minutos ou mais tarde a partir da hora atual.
- Posição relativa: escolha onde inserir o programa na programação em relação a outro programa. Você pode selecionar Antes do programa ou Depois do programa. Essa configuração não se aplica se esse for o primeiro programa na programação do seu canal.
- Programa relativo: o nome do programa a ser usado para inserir o novo programa antes ou depois. Essa configuração não se aplica se esse for o primeiro programa na programação do seu canal.

Se você escolher Selecionar um programa existente, selecione o nome do programa em uma lista predefinida dos próximos 100 programas reproduzidos pelo canal no menu suspenso Usar programa existente.

Se você escolher Pesquisar um programa pelo nome, insira o nome de um programa existente em seu canal.

Se você quiser adicionar intervalos publicitários ao seu programa, vá para a próxima etapa. Os intervalos publicitários só podem ser configurados para programas que usam fontes de VOD. Para fontes ativas, os intervalos de anúncios nos manifestos do DASH e os intervalos nos manifestos do HLS que usam a EXT-X-DATERANGE tag são transmitidos automaticamente.

14. Selecione Adicionar intervalo publicitário. Em Intervalos publicitários, defina as configurações do intervalo publicitário:

- Nome do local de origem do Slate: Escolha Selecionar um local de origem existente e escolha o local de origem em que o Slate está armazenado, criado anteriormente neste tutorial.
- Nome da fonte de VOD: escolha Selecionar uma fonte de VOD existente e escolha a fonte de VOD que você está usando para o slate que você adicionou anteriormente neste tutorial. A duração da lista determina a duração do intervalo publicitário.
- Para deslocamento em milissegundos: esse valor determina o horário de início do intervalo publicitário em milissegundos, como um deslocamento em relação ao início do programa. Insira qualquer valor menor que a duração da fonte de VOD e que esteja alinhado com um limite de segmento em todas as faixas da fonte VOD do programa (todas as faixas de áudio, vídeo e legendas ocultas), caso contrário, o intervalo do anúncio será ignorado. Por exemplo,

se você inserir 0, isso criará um intervalo publicitário pré-lançamento que será reproduzido antes do início do programa.

 Note

Se MediaTailor detectar marcadores de anúncios, como DATERANGE ou EXT-X-CUE-OUT para HLS e EventStream para DASH, com durações de zero em sua fonte de VOD, você poderá selecionar o deslocamento desses marcadores de anúncio no menu suspenso para serem usados como compensação do intervalo publicitário. Para que uma oportunidade de anúncio seja detectada, ela deve estar presente com o mesmo deslocamento em todas as configurações de pacote em uma fonte de VOD e sua duração deve ser zero.

- Tipo de mensagem: O tipo de inserção de anúncios SCTE-35. Escolha SPLICE_INSERT ou TIME_SIGNAL:
 - SPLICE_INSERT: fornece metadados básicos sobre o intervalo do anúncio usando parâmetros de inserção de emenda.
 - TIME_SIGNAL: fornece metadados mais avançados usando descritores de segmentação. Para obter mais informações sobre as diferenças entre os tipos de mensagens, consulte [the section called “SCTE-35 mensagens para pausas publicitárias”](#).

Para o tipo de mensagem SPLICE_INSERT:

- Para o número Avail, ele é gravado `splice_insert.avail_num`, conforme definido na seção 9.7.3.1. da especificação SCTE-35. O valor padrão é 0. Os valores devem estar entre 0 e 256, inclusivos.
- Para o Avail esperado, isso é gravado `splice_insert.avails_expected`, conforme definido na seção 9.7.3.1. da especificação SCTE-35. O valor padrão é 0. Os valores devem estar entre 0 e 256, inclusivos.
- Para o ID do evento Splice, isso é gravado `splice_insert.splice_event_id`, conforme definido na seção 9.7.3.1. da especificação SCTE-35. O valor padrão é 1.
- Para ID de programa exclusivo, isso é gravado `splice_insert.unique_program_id`, conforme definido na seção 9.7.3.1. da especificação SCTE-35. O valor padrão é 0. Os valores devem estar entre 0 e 256, inclusivos.

Para o tipo de mensagem TIME_SIGNAL:

- Para ID do evento de segmentação, isso é `gravadosegmentation_descriptor.segmentation_event_id`, conforme definido na seção 10.3.3.1 da especificação SCTE-35. O valor padrão é 1.
- Para ID do tipo de segmentação, isso é `gravadosegmentation_descriptor.segmentation_type_id`, conforme definido na seção 10.3.3.1 da especificação SCTE-35. O valor padrão é 48 (0x30). Os valores devem estar entre 0 e 256, inclusivos.
- Para segmentação UPID, isso é `gravadosegmentation_descriptor.segmentation_upid`, conforme definido na seção 10.3.3.1 da especificação SCTE-35. O valor deve ser uma string hexadecimal contendo caracteres e. 0-9 A-F O valor padrão é uma string vazia.
- Para o tipo UPID de segmentação, isso é `gravadosegmentation_descriptor.segmentation_upid_type`, conforme definido na seção 10.3.3.1 da especificação SCTE-35. O valor padrão é 14 (0x0E). Os valores devem estar entre 0 e 256, inclusivos.
- Para o número do segmento, isso é `gravadosegmentation_descriptor.segment_num`, conforme definido na seção 10.3.3.1 da especificação SCTE-35. O valor padrão é 0. Os valores devem estar entre 0 e 256, inclusivos.
- Para segmentos esperados, isso é `gravadosegmentation_descriptor.segments_expected`, conforme definido na seção 10.3.3.1 da especificação SCTE-35. O valor padrão é 0. Os valores devem estar entre 0 e 256, inclusivos.

15. Escolha Adicionar programa.

Para obter informações mais avançadas sobre MediaTailor como personalizar seus intervalos publicitários, consulte [Insira anúncios e intervalos de anúncios personalizados em um stream do canal](#).

Note

Se seu canal tiver pelo menos uma saída com um tipo de marcação de Enhanced Scte35 anúncio, você poderá enviar metadados de ad-break. MediaTailor grava os pares de valores-chave enviados na EXT-X-ASSET tag para seu intervalo publicitário.

Definindo coortes de público e conteúdo alternativo com as Regras do Programa

Com as regras do programa, você pode definir coortes de público para um canal e especificar mídias alternativas a serem reproduzidas para esses públicos. Você pode associar uma ou mais fontes alternativas de conteúdo a um público para um programa. Depois que o programa terminar, o conteúdo padrão do público será reproduzido, a menos que você especifique outras mídias alternativas.

As regras do programa estão disponíveis nos canais de nível STANDARD com o modo de reprodução LINEAR. MediaTailor canais suportam mídia alternativa para todas as fontes de VOD e fontes ao vivo.

Para ver um exemplo de uso, consulte [Usando as regras do programa com AWS MediaTailor](#).

Definindo públicos

Defina o público em um canal digitando o público um por um ao configurar um canal. MediaTailor Você pode fazer isso por meio do MediaTailor console ou da MediaTailor CreateChannel API. Cada público deve ter entre 1 e 32 caracteres alfanuméricos. Se os valores fornecidos para os públicos forem inválidos, a solicitação falhará.

Você só pode definir públicos em canais de nível STANDARD com o modo de reprodução LINEAR.

Quando precisar atualizar os públicos, você pode fazer isso usando o MediaTailor console ou a MediaTailor UpdateChannel API.

Se você estiver usando o ProgramRules recurso, certifique-se de que o AudienceMedia definido em CreateProgram ou a UpdateProgram solicitação contenha o público existente definido no canal.

Criação de mídia alternativa

A tarefa a seguir explica como definir mídia alternativa usando o MediaTailor console. Para obter informações sobre como definir mídia alternativa usando a MediaTailor API, consulte [CreateProgram](#) Referência da AWS Elemental MediaTailor API.

Para definir mídia alternativa em um novo programa:

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, selecione Montagem de canais > Canais.

3. Selecione o nome do canal ao qual você deseja adicionar mídia alternativa.
4. Crie um programa. Para obter mais informações, consulte [Criação de um programa dentro da programação de um canal usando o MediaTailor console](#).
5. Configure a mídia alternativa:
 - Selecione Adicionar na caixa Públicos para selecionar o público para o qual você está definindo mídia alternativa.
 - Selecione um público definido no canal no menu Público.
 - Selecione Adicionar mídia alternativa para começar a definir a mídia alternativa para o programa.
 - MediaTailor cria uma caixa de mídia alternativa 1. Esse é o primeiro conteúdo que é MediaTailor reproduzido como mídia alternativa no programa.
 - Dentro da caixa Mídia alternativa 1:
 - Selecione um local de origem.
 - Selecione um tipo de VOD ou fonte ao vivo:

Para VOD

- Selecione VOD para o tipo de fonte.
- (Opcional) especifique um intervalo de clipes. Com fontes de VOD, incluindo fontes VOD de mídia alternativa, você pode especificar uma parte de uma fonte de VOD para reproduzir, recortando do início ao fim da fonte. and/or Especifique Os deslocamentos inicial e final estão em milissegundos.
- (Opcional) a\ Adicionar intervalos publicitários. Isso é feito da mesma forma que ao criar programas. Para obter mais informações, consulte [Criação de um programa dentro da programação de um canal usando o MediaTailor console](#).

Para ao vivo

- Selecione Ao vivo para o tipo de fonte.
- Selecione uma fonte ao vivo.
- Insira um horário de início em milissegundos do horário em que essa fonte ativa deve iniciar. A fonte ao vivo só será reproduzida dentro do período de tempo do programa padrão em que está sendo definida. Se a hora de início for anterior ao início do programa padrão, ela não começará até que o programa padrão comece. Se a hora de início for após o término do programa padrão, não MediaTailor reproduzirá a fonte ao vivo.

- Insira uma Duração em milissegundos. A duração deve ser de pelo menos 10 minutos.
- Mídia alternativa adicional pode ser adicionada a esse programa para o público selecionando Adicionar mídia alternativa novamente. Isso criará outra caixa chamada Mídia alternativa 2. Você pode especificar até 5 fontes de mídia alternativa por programa, por público.
- Quando terminar de definir a mídia alternativa para todos os públicos desejados, selecione Avançar e continue criando o programa.

Para obter mais informações, consulte [Criação de um programa dentro da programação de um canal usando o MediaTailor console](#).

Note

A mídia alternativa só é reproduzida no período de tempo do programa em que está definida. Se todo o conteúdo alternativo ultrapassar o conteúdo padrão, MediaTailor ele será truncado. MediaTailor reproduz mídia alternativa na ordem em que está definida. Os horários de início da mídia alternativa ao vivo sempre terão precedência e truncarão as fontes de VOD ou fontes ao vivo previamente programadas. Qualquer momento que não seja preenchido com mídia alternativa para um público será preenchido com a lista de preenchimento definida pelo canal

- Para definir a mídia de público para outros públicos, selecione Adicionar novamente ao lado de Públicos. Selecione o público recém-criado, defina o ID do público e adicione mídia alternativa conforme descrito acima. Até 5 públicos podem ter mídias alternativas em qualquer programa.

Gerando manifestos específicos para o público

Para recuperar um manifesto para um público específico, use o parâmetro de `aws.mediatailor.channel.audienceId` consulta. Esse parâmetro de consulta pode ser anexado dinamicamente pela sua CDN ou adicionado por meio de uma chamada ao seu conteúdo ou sistema de gerenciamento de clientes. Você deve manter a associação de uma determinada sessão de reprodução a uma parte `audienceId` externa da MediaTailor. Isso recuperará um manifesto específico do público com qualquer mídia alternativa definida para esse público no lugar do conteúdo padrão. É importante que, quando um manifesto for solicitado para um público

específico, o player sempre solicite o manifesto com o mesmo ID de público, caso contrário poderão ocorrer erros de reprodução.

Se uma solicitação for feita para um público que não existe no canal, MediaTailor retornará um erro 404.

Example Obtendo um manifesto para o público

```
https:// prefix>.channel-assembly.mediatailor.us-  
west-2.amazonaws.com/v1/channel/ExampleChannel/index_dash.mpd?  
aws.mediatailor.channel.audienceId=Seattle
```

Insira anúncios e intervalos de anúncios personalizados em um stream do canal

Com MediaTailor, você pode monetizar fluxos lineares de montagem de canais inserindo intervalos publicitários em seus programas sem condicionar o conteúdo com marcadores. SCTE-35 Você pode usar a montagem de canais com o serviço de inserção de MediaTailor anúncios ou com qualquer inserção de anúncios do lado do servidor (SSAI).

Os tópicos a seguir mostram como inserir anúncios personalizados e intervalos publicitários no fluxo linear do seu canal.

Tópicos

- [Configurando a inserção de anúncios com MediaTailor](#)
- [SCTE-35 mensagens para pausas publicitárias](#)

Configurando a inserção de anúncios com MediaTailor

Para inserir anúncios personalizados no stream do seu canal, o URL do endpoint do seu canal é a fonte de conteúdo para AWS Elemental MediaTailor. Este guia mostra como configurar MediaTailor a inserção de anúncios.

Pré-requisitos

Antes de começar, certifique-se de que os seguintes requisitos são atendidos:

- Prepare seus streams HLS e DASH para MediaTailor inserção de anúncios.

- Se você ainda não preparou fluxos de conteúdo, consulte o [Etapa 2: preparar um stream](#) tópico Introdução à inserção de MediaTailor anúncios.
- Tenha um servidor de decisão de anúncios (ADS).
- Defina as configurações do Ad Break no programa. Para obter mais informações, consulte o [Configuring ad breaks for your program](#) procedimento.

Como prática recomendada, considere usar uma rede de distribuição de conteúdo (CDN) entre a montagem do canal e a inserção do MediaTailor anúncio. O serviço de inserção de MediaTailor anúncios pode gerar solicitações de origem adicionais. Portanto, é uma prática recomendada configurar sua CDN para fazer proxy dos manifestos da montagem do canal e, em seguida, usar as URLs prefixadas da CDN na URL da fonte de conteúdo.

Configurar MediaTailor para inserção de anúncios

Veja a seguir como definir as configurações do MediaTailor console para que você possa inserir anúncios personalizados no stream do seu canal.

Para configurar MediaTailor para inserção de anúncios

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Configurações.
3. Em Configurações obrigatórias, insira as informações básicas necessárias sobre sua configuração:
 - Nome: O nome da sua configuração.
 - Fonte de conteúdo: insira o URL de reprodução da saída do seu canal, menos o nome do arquivo e a extensão. Para obter informações avançadas sobre MediaTailor configuração, consulte [Configurações necessárias](#).
 - Servidor de decisão de anúncios: insira o URL do seu ADS.
4. Opcionalmente, você pode definir os aliases de configuração, os detalhes de personalização e as configurações avançadas. Para obter informações sobre essas configurações, consulte [Configurações opcionais](#).
5. Na barra de navegação, escolha Criar configuração.

Agora que você configurou MediaTailor a inserção de anúncios, também pode configurar intervalos para anúncios. Para obter instruções detalhadas, consulte [Introdução à inserção MediaTailor de anúncios](#).

SCTE-35 mensagens para pausas publicitárias

Com MediaTailor, você pode criar um canal de conteúdo com base na localização da fonte e nos recursos de origem do VOD. Em seguida, você pode configurar um ou mais intervalos publicitários para cada um dos programas na programação de um canal. Você usa mensagens com base na SCTE-35 especificação para condicionar o conteúdo para intervalos publicitários. Por exemplo, você pode usar SCTE-35 mensagens para fornecer metadados sobre os intervalos publicitários. Para obter mais informações sobre a SCTE-35 especificação, consulte [Mensagem de sinalização de inserção de programa digital](#).

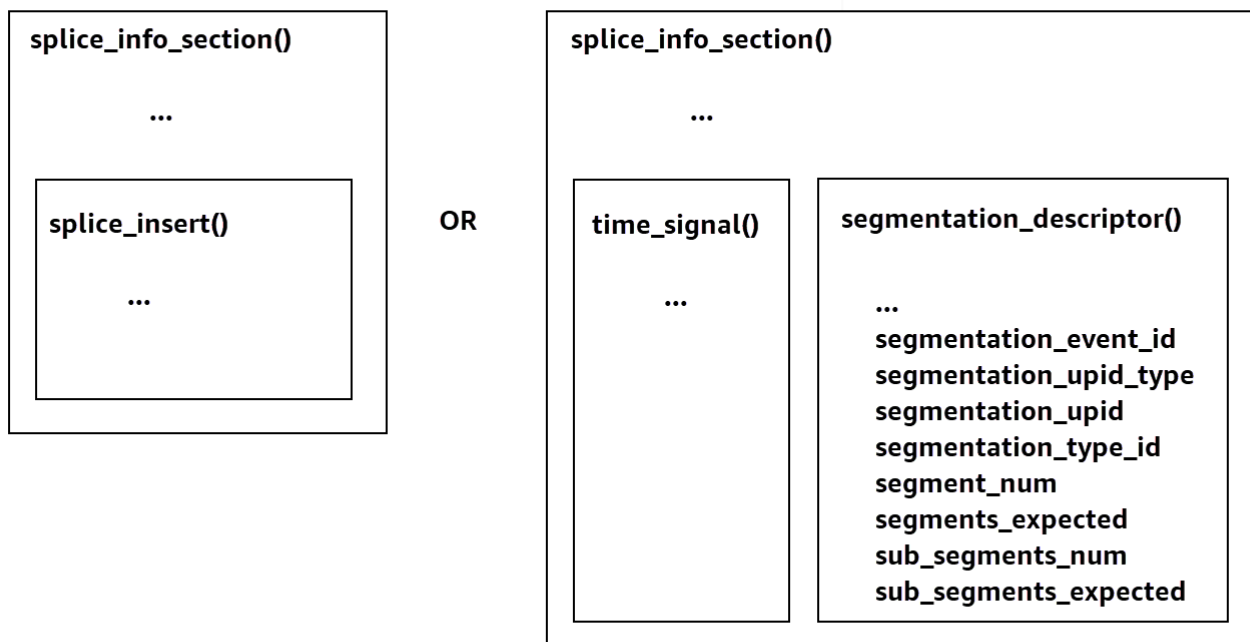
Você configura os intervalos publicitários de duas maneiras:

- Anexando uma `time_signal` SCTE-35 mensagem com uma `segmentation_descriptor` mensagem. Essa `segmentation_descriptor` mensagem contém campos de metadados mais avançados, como identificadores de conteúdo, que transmitem mais informações sobre o intervalo do anúncio. MediaTailor grava os metadados do anúncio no manifesto de saída como parte dos dados do marcador de anúncio EXT-X-DATERANGE (HLS) ou EventStream (DASH). SCTE-35
- Anexar uma `splice_insert` SCTE-35 mensagem que forneça metadados básicos sobre o intervalo publicitário.
- HLS:
 - Quando o tipo de marcação do anúncio é `Daterange`, MediaTailor especifica quebras de anúncio como EXT-X-DATERANGE tags no manifesto.
 - Quando o tipo de marcação do anúncio é `Scte35 Enhanced`, MediaTailor especifica quebras de anúncio usando as seguintes tags:
 - MediaTailor coloca um anúncio EXT-X-CUE-OUT no primeiro segmento da lista de anúncios, indicando um corte do conteúdo até o intervalo do anúncio. Ele contém a duração esperada do intervalo publicitário, como EXT-X-CUE-OUT:Duration=30.
 - >EXT-X-ASSET: Essa tag aparece no mesmo segmento EXT-X-CUE-OUT e contém os metadados do ad-break fornecidos no AdBreak quando o programa é criado ou atualizado. Sempre contém CAID.
 - EXT-0ATCLS-SCTE35: essa tag aparece no mesmo segmento EXT-X-CUE-OUT e contém bytes codificados em base64 da mensagem. SCTE-35

- **EXT-X-CUE-OUT-CONT**: essa tag aparece em cada segmento subsequente na lista de anúncios e contém informações sobre duração e tempo decorrido. Ele também contém a SCTE-35 mensagem codificada em base64 e o CAID
- **EXT-X-CUE-IN**: essa tag aparece no primeiro segmento do conteúdo após o término do intervalo publicitário e indica um corte de um intervalo publicitário de volta ao conteúdo.

A ilustração a seguir mostra as duas maneiras de configurar intervalos publicitários em um canal usando SCTE-35 mensagens:

- Use uma `splice_insert()` mensagem para configurar intervalos publicitários com metadados básicos.
- Use uma `time_signal()` mensagem junto com uma `segmentation_descriptor()` mensagem para configurar intervalos publicitários com metadados mais detalhados.



Para obter informações sobre o `time_signal`, consulte a seção 9.7.4 da SCTE-35 especificação 2022, Mensagem de sinalização de [inserção de programa digital](#).

As informações do intervalo publicitário aparecem nos `splice_info_section` SCTE-35 dados de saída. Com MediaTailor, você pode emparelhar uma única `segmentation_descriptor` mensagem com uma única `time_signal` mensagem.

Note

Se você enviar uma `segmentation_descriptor` mensagem, deverá enviá-la como parte do tipo de `time_signal` mensagem. A `time_signal` mensagem contém somente o `splice_time` campo que é MediaTailor construído usando um determinado timestamp.

A tabela a seguir descreve os campos MediaTailor necessários para cada `segmentation_descriptor` mensagem. Para obter mais informações, consulte a seção 10.3.3.1 da SCTE-35 especificação 2022, que você pode adquirir no site da [ANSI Webstore](#).

Campos obrigatórios para uma mensagem **`segmentation_descriptor`**

Campo	Tipo	Valor padrão	Description
<code>segmentation_event_id</code>	integer	1	Isso foi escrito para <code>segmentation_descriptor.segmentation_event_id</code> .
<code>segmentation_upid_type</code>	integer	14 (0x0E)	Isso foi escrito para <code>segmentation_descriptor.segmentation_upid_type</code> . O valor deve estar entre 0 e 256, inclusive.
<code>segmentation_upid</code>	string	"" (string vazia)	Isso foi escrito para <code>segmentation_descriptor</code> .

Campo	Tipo	Valor padrão	Description
			mentation_upid . O valor deve ser uma string hexadecimal, contendo caracteres e. 0-9 A-F
segmentat ion_type_id	integer	48 (0x30)	Isso foi escrito parasegmentat ion_descr iptor.seg mentation _type_id . O valor deve estar entre 0 e 256, inclusive.
segment_num	integer	0	Isso foi escrito parasegmentat ion_descr iptor.seg ment_num . O valor deve estar entre 0 e 256, inclusive.
segments_ expected	integer	0	Isso foi escrito parasegmentat ion_descr iptor.seg ments_expected . O valor deve estar entre 0 e 256, inclusive.

Campo	Tipo	Valor padrão	Description
sub_segment_num	integer	null	Isso foi escrito parasegmentation_descriptor.sub_segment_num . O valor deve estar entre 0 e 256, inclusive.
sub_segments_expected	integer	null	Isso foi escrito parasegmentation_descriptor.sub_segments_expected . O valor deve estar entre 0 e 256, inclusive.

A tabela a seguir mostra os valores MediaTailor definidos automaticamente para alguns dos campos da segmentation_descriptor mensagem.

Valores definidos por MediaTailor para os campos de uma mensagem

segmentation_descriptor

Campo	Tipo	Valor
segmentation_event_cancel_indicator	Booleano	Verdadeiro
program_segmentation_flag	Booleano	Verdadeiro
delivery_not_restricted_flag	Booleano	Verdadeiro

MediaTailor sempre define o `segmentation_duration_flag` valor `True`. MediaTailor preenche o `segmentation_duration` campo com a duração, em tiques, do conteúdo do estado.

 Note

Ao MediaTailor enviar as `time_signal` mensagens, ele define o `splice_command_type` campo na `splice_info_section` mensagem como 6 (0x06).

Na saída HLS, para um AdBreak com uma `time_signal` mensagem, a EXT-X-DATERANGE tag de saída inclui um SCTE-35 campo definido para a versão serializada da `splice_info_section` mensagem. Por exemplo, a EXT-X-DATERANGE tag a seguir mostra a versão serializada da `splice_info_section` mensagem:

```
#EXT-X-DATERANGE:ID="\1",START-DATE="\2020-09-25T02:13:20Z",DURATION=300.0,SCTE35-OUT=0xFC002C0000000000000000FF05068000000000001602144355454900000000100E000019BFCC00E00300000000
```

Na saída DASH, para um AdBreak com uma `time_signal` mensagem, o EventStream elemento de saída inclui um `scte35:SpliceInfoSection` elemento com `scte35:TimeSignal` e `scte35:SegmentationDescriptor` elementos como seus filhos. O `scte35:TimeSignal` elemento tem um `scte35:SpliceTime` elemento filho e o `scte35:SegmentationDescriptor` elemento tem um `scte35:SegmentationUpid` elemento filho. Por exemplo, a saída DASH a seguir mostra a estrutura do EventStream elemento filho do elemento:

```
<EventStream schemeIdUri="urn:scte:scte35:2013:xml" timescale="900000">
  <Event duration="270000000">
    <scte35:SpliceInfoSection protocolVersion="0" ptsAdjustment="0" tier="4095">
      <scte35:TimeSignal>
        <scte35:SpliceTime ptsTime="0" />
      </scte35:TimeSignal>
      <scte35:SegmentationDescriptor segmentNum="0"
segmentationDuration="270000000" segmentationEventCancelIndicator="false"
segmentationEventId="1" segmentationTypeId="48" segmentsExpected="0">
        <scte35:SegmentationUpid segmentationUpidFormat="hexBinary"
segmentationUpidType="14">012345</scte35:SegmentationUpid>
      </scte35:SegmentationDescriptor>
    </scte35:SpliceInfoSection>
  </Event>
</EventStream>
```

Você aprendeu sobre o uso de SCTE-35 mensagens para configurar intervalos publicitários na montagem de canais, a estrutura e os campos obrigatórios dessas mensagens e exemplos de saída HLS e DASH que incluem as SCTE-35 mensagens.

Mudando o horário da reprodução de um canal

Com a montagem de MediaTailor canais, você pode mudar o horário da reprodução de um canal. A mudança de horário permite que os espectadores assistam ao conteúdo de um canal em um horário diferente do horário original da transmissão. Por exemplo, um espectador pode começar a assistir a um programa desde o início, mesmo que participe da transmissão após o início. Ou um espectador pode pausar um programa e continuar assistindo mais tarde.

Para mudar o horário da reprodução de um canal, você adiciona um `start` parâmetro ao URL de reprodução do canal. O `start` parâmetro especifica a hora em que a reprodução deve começar, em relação à hora atual. Por exemplo, se o horário atual for 14h e você quiser iniciar a reprodução a partir das 13h, defina o `start` parâmetro como -1 hora em segundos. -3600

O `start` parâmetro aceita valores positivos e negativos, em segundos:

- Valores negativos indicam uma hora no passado, em relação à hora atual. Por exemplo, `start=-3600` significa “iniciar a reprodução há 1 hora”.
- Valores positivos indicam uma hora no futuro, em relação à hora atual. Por exemplo, `start=3600` significa “iniciar a reprodução a partir de 1 hora no futuro”.

O exemplo a seguir mostra como adicionar o `start` parâmetro ao URL de reprodução de um canal:

```
https://a1b2c3d4e5f6.mediapackage.us-west-2.amazonaws.com/out/v1/examplechannel/  
index.m3u8?start=-3600
```

Neste exemplo, a reprodução começa há 1 hora.

O `start` parâmetro está sujeito às seguintes limitações:

- A janela máxima de mudança de horário é de 24 horas no passado ou no futuro.
- A mudança de horário só está disponível para canais que usam o modo de reprodução linear.
- A mudança de horário não está disponível para canais que usam o modo de reprodução em loop.

Se você especificar um `start` parâmetro que esteja fora da janela de mudança de horário disponível, MediaTailor retornará um erro.

Usando uma CDN para otimizar a personalização de MediaTailor anúncios e a entrega de conteúdo

AWS Elemental MediaTailor funciona de forma eficaz como um serviço independente, mas integrá-lo a uma rede de distribuição de conteúdo (CDN), como a Amazon CloudFront ou outros terceiros CDNs, pode melhorar significativamente seus fluxos de trabalho de streaming. Uma integração de CDN é particularmente valiosa quando você precisa veicular conteúdo para um público grande e distribuído geograficamente ou quando deseja garantir a entrega consistente de anúncios em diferentes regiões. AWS

Sem uma CDN, seus espectadores se conectam diretamente MediaTailor para manifestos e segmentos de anúncios personalizados, o que pode aumentar a latência, especialmente para espectadores localizados longe da AWS região em que sua MediaTailor configuração está implantada. Além disso, durante eventos de alto tráfego, as conexões diretas MediaTailor podem aumentar a carga, potencialmente afetando o desempenho.

Para obter mais informações sobre MediaTailor conceitos e fluxos de trabalho, consulte [O que é AWS Elemental MediaTailor?](#).

Ao integrar uma CDN MediaTailor, é importante configurar o tratamento adequado do CORS (Cross-Origin Resource Sharing) para evitar problemas que possam causar falhas de reprodução em players baseados na web. A configuração correta do CORS é essencial tanto para segmentos de anúncios quanto para segmentos de conteúdo. Embora os segmentos de anúncios sejam mais suscetíveis a problemas de CORS, aplicar o tratamento consistente do CORS em todos os tipos de segmentos garante a experiência de reprodução mais confiável. Para obter orientações detalhadas sobre como configurar os comportamentos de roteamento de CDN com o tratamento adequado do CORS, consulte. [Production-ready CloudFront configuração para MediaTailor](#)

A integração com a CDN também permite recursos avançados de passagem de parâmetros e roteamento dinâmico. Para obter informações sobre como transmitir parâmetros de consulta CDNs para autorização e roteamento, consulte [MediaTailor parâmetros de consulta do manifesto](#). Para ver o servidor de anúncios dinâmico e o roteamento de origem usando aliases de configuração, consulte. [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#)

Colocar uma CDN entre seus espectadores MediaTailor oferece os seguintes benefícios:

- Reduza a latência veiculando conteúdo de locais periféricos mais próximos dos espectadores

- Melhore a escalabilidade distribuindo a carga em toda a infraestrutura global da CDN
- Aumente a confiabilidade por meio de caminhos de entrega redundantes
- Otimize os custos reduzindo o tráfego de origem
- Implemente recursos avançados como o Media Quality-Aware Routing (MQAR) para melhorar a qualidade de streaming

Tópicos

- [Selecione a CDN certa para suas necessidades](#)
- [Planeje sua integração de CDN para AWS Elemental MediaTailor](#)
- [Configure a integração da CDN com MediaTailor](#)
- [Configure o SSAI com uma CDN para publicidade em vídeo personalizada](#)
- [Crie canais MediaTailor lineares com montagem de canais e CDN](#)
- [Integre MediaTailor com uma MediaPackage CDN](#)
- [Integração AWS Elemental MediaTailor com a Amazon CloudFront](#)
- [Configure terceiros CDNs para entrega MediaTailor de anúncios](#)
- [Guia de otimização de desempenho para CDN e integrações MediaTailor](#)
- [Monitore as operações e o desempenho da MediaTailor CDN](#)
- [Teste e validação para CDN e integrações MediaTailor](#)
- [Solucionar problemas com a integração com MediaTailor a CDN](#)
- [Análise de log de integração de CDN e referência de código de erro para MediaTailor](#)
- [Automatize MediaTailor e CDN com CloudFormation](#)
- [Production-ready CloudFront configuração para MediaTailor](#)
- [Obtenha suporte e ajuda para solução de problemas para CDN e integrações MediaTailor](#)

Selecione a CDN certa para suas necessidades

Escolher o provedor certo de rede de distribuição de conteúdo (CDN) é uma decisão importante que pode afetar o desempenho, o custo e a experiência do espectador na entrega de AWS Elemental MediaTailor conteúdo. Considere esses fatores ao selecionar uma CDN para sua MediaTailor implementação:

Cobertura geográfica

Escolha uma CDN com forte presença nas regiões onde seu público está localizado. Diferentes provedores de CDN têm pontos fortes variados em diferentes regiões geográficas.

Integração com AWS serviços

A Amazon CloudFront oferece a mais estreita integração com MediaTailor outros AWS serviços, o que pode simplificar a configuração e o gerenciamento. Terceiros CDNs podem oferecer outras vantagens, como recursos especializados de entrega de vídeo ou presença mais forte em regiões específicas.

Recursos específicos de vídeo

Procure esses recursos de CDNs oferta projetados especificamente para entrega de vídeo, como otimização adaptativa de taxa de bits, compactação de vídeo e análises focadas na experiência do espectador.

Estrutura de custos

Compare modelos de preços entre provedores, considerando fatores como volume de tráfego, distribuição geográfica e requisitos de recursos. Alguns CDNs oferecem descontos por volume ou descontos por uso contínuo que podem estar de acordo com seus padrões de uso.

Support para recursos avançados

Verifique se a CDN escolhida oferece suporte aos recursos de que você precisa, como autenticação de token, restrição geográfica, colapso de solicitações e encaminhamento adequado do cabeçalho.

Para obter mais informações sobre provedores de CDN específicos e sua integração com MediaTailor, consulte os recursos a seguir.

- [Integração do CloudFront](#) para Amazon CloudFront
- [Configuração de CDN de terceiros](#) para provedores de CDN terceirizados

Os tópicos a seguir fornecem orientação abrangente sobre a configuração MediaTailor com uma CDN para um desempenho ideal.

Planeje sua integração de CDN para AWS Elemental MediaTailor

Você pode melhorar a experiência do espectador e reduzir a latência com uma integração de CDN para AWS Elemental MediaTailor. Ao implementar uma rede de distribuição de conteúdo (CDN), você pode fornecer conteúdo de locais mais próximos aos seus espectadores. Isso garante tempos de carregamento mais rápidos, melhor escalabilidade e entrega consistente de anúncios em diferentes regiões geográficas.

Você precisa de um planejamento adequado antes de implementar uma CDN com AWS Elemental MediaTailor. Esta seção orienta você pelas principais áreas de planejamento. Você aborda essas áreas antes de iniciar a configuração real. Essas etapas ajudam você a criar uma experiência de visualização ideal para seu público.

Depois de concluir seu planejamento, consulte [Integrando uma CDN com MediaTailor para obter instruções](#) de step-by-step implementação.

Para obter informações sobre MediaTailor cotas que podem afetar seu planejamento de CDN, consulte [Cotas em AWS Elemental MediaTailor](#). Para obter informações sobre CloudFront cotas, consulte [Cotas](#) no Guia do CloudFront desenvolvedor.

Antes de começar a planejar, entenda como MediaTailor interage com uma CDN:

1. Os espectadores solicitam conteúdo por meio de sua CDN, em vez de diretamente de MediaTailor.
2. A CDN encaminha solicitações manifestas MediaTailor para personalização.
3. A CDN armazena em cache e veicula segmentos de conteúdo e segmentos de anúncios a partir de pontos de presença.

Essa arquitetura reduz a carga MediaTailor enquanto você garante que os espectadores recebam anúncios personalizados com latência mínima.

Compreender a terminologia do manifesto ajuda você a configurar sua CDN corretamente.

Protocolos de streaming diferentes usam estruturas de manifesto específicas que afetam a forma como você configura o armazenamento em cache e o roteamento:

- Manifestos HLS - Quando você trabalha com fluxos HLS, você lida com:
 - Playlist multivariante: configure sua CDN para encaminhar esses manifestos de alto nível para personalização. MediaTailor

- **Playlist de mídia:** defina as regras de armazenamento em cache apropriadas para esses manifestos que contêm links para segmentos de conteúdo.
- **Manifestos do DASH** - Quando você trabalha com fluxos do DASH, você lida com:
 - **MPD (Descrição da apresentação de mídia):** configure sua CDN para lidar com esses manifestos de acordo com seus requisitos de personalização.

O processo de planejamento da CDN envolve essas etapas principais, cada uma focada em uma tarefa específica:

- [Estime os requisitos de tráfego para CDN e integrações MediaTailor](#): calcule a simultaneidade esperada do espectador e os requisitos de largura de banda.
- [Configure estratégias de otimização para CDN e integrações MediaTailor](#): configure sua CDN para oferecer conteúdo e personalização de anúncios otimizados.
- [Personalize o planejamento para CDN e integrações MediaTailor](#): ajuste sua estratégia de CDN com base em seu MediaTailor fluxo de trabalho específico.
- [Configure o monitoramento e o escalonamento para CDN e integrações MediaTailor](#): Implemente estratégias de monitoramento e escalonamento para um desempenho confiável.
- [Otimize os custos de CDN e integrações MediaTailor](#): Equilibre o desempenho com a eficiência de custos.
- [Teste sua implementação para CDN e MediaTailor integrações](#): teste minuciosamente sua integração com a CDN antes da implantação na produção.

Estime os requisitos de tráfego para CDN e integrações MediaTailor

Para dimensionar com precisão sua integração de rede de distribuição de conteúdo (CDN) com AWS Elemental MediaTailor:

1. Calcule a simultaneidade esperada do espectador usando dados históricos ou eventos semelhantes. Planeje uma capacidade adicional além de sua linha de base para lidar com picos inesperados. Para obter recomendações de escalabilidade atuais, consulte sua equipe de AWS contas. Você também pode ver [Cotas em AWS Elemental MediaTailor](#).
2. Identifique padrões de pico de tráfego e possíveis picos em sua programação de conteúdo. Considere fatores como:
 - Eventos esportivos ao vivo ou estreias de temporada

- Campanhas de marketing ou eventos promocionais
 - Diferenças de fuso horário para públicos globais
 - Padrões de visualização de feriados ou sazonais
3. Determine seus requisitos de largura de banda multiplicando a contagem de espectadores pelas taxas de bits do stream. Trabalhe com seu provedor de CDN para calcular a capacidade apropriada. Baseie esse cálculo nas taxas de bits de seu conteúdo específico e no tamanho esperado do público. Adicione sobrecarga para segmentos de anúncios e solicitações de manifesto conforme recomendado pelo seu provedor.
 4. Trabalhe com seu provedor de CDN para garantir capacidade de borda suficiente em suas regiões-alvo.

Garanta que sua capacidade de inserção de anúncios atenda à demanda dos espectadores realizando estas ações específicas:

1. Verifique sua cota atual de solicitações de inserção de anúncios no console [Service Quotas](#). Analise os limites atuais do serviço para entender quantos visualizadores simultâneos sua configuração pode suportar.
2. Para eventos de alto tráfego, solicite uma cota maior por meio do console [Service Quotas](#).
3. Se você espera mais de 500.000 espectadores simultâneos, entre em contato com o AWS [Support](#) pelo menos duas semanas antes do evento. Isso permite AWS garantir capacidade suficiente para suas necessidades de personalização de anúncios.

Para obter mais informações sobre a implementação do planejamento de capacidade em seu fluxo de trabalho, consulte [Usando o agendamento de pré-busca](#) para otimizar a entrega de anúncios para eventos de alto tráfego.

Configure estratégias de otimização para CDN e integrações MediaTailor

Ao concluir sua estimativa de tráfego, configure sua rede de entrega de conteúdo (CDN) para otimizar a entrega de conteúdo e a personalização de anúncios com AWS Elemental MediaTailor. Essas otimizações ajudam a garantir uma reprodução suave, mantendo a publicidade direcionada.

Implemente essas otimizações específicas de CDN que seguem:

1. Configure os recursos do Origin Shield em sua CDN para reduzir a carga MediaTailor e melhorar a eficiência do armazenamento em cache. O Origin Shield atua como uma camada intermediária de cache que:

- Consolide várias solicitações de visualizadores em uma única solicitação de origem
- Reduza o número de solicitações redundantes para MediaTailor
- Melhore os tempos de resposta do conteúdo em cache

Para obter detalhes de implementação sobre como configurar o Origin Shield com CloudFront, consulte [Usando o Origin Shield](#) no Guia do CloudFront Desenvolvedor.

2. Defina valores apropriados de Time To Live (TTL) para diferentes tipos de conteúdo. O TTL determina por quanto tempo o CDN armazena o conteúdo em cache. Após esse período, o CDN solicita uma nova cópia da origem:

- Manifestos:
 - 10 segundos para inserção de anúncios
 - 5-10 segundos para montagem do canal

Na inserção de anúncios, MediaTailor fornece manifestos com anúncios personalizados para o espectador. Se uma playlist ou MPD estiver em cache e exibida no dispositivo de reprodução errado, o dispositivo poderá encontrar problemas de reprodução ou rastreamento.

- Segmentos de conteúdo: 24 horas ou mais (eles raramente mudam e você pode armazená-los em cache agressivamente para reduzir a carga de origem)
- Segmentos de anúncios: 24 horas ou mais (o conteúdo do anúncio geralmente é reutilizado entre os espectadores e você pode armazená-lo em cache por longos períodos)

Para recomendações abrangentes de TTL e estratégias de otimização de cache em todos os MediaTailor fluxos de trabalho, consulte [Otimização de cache para CDN e integrações MediaTailor](#)

Para obter instruções detalhadas sobre como configurar comportamentos de cache em CloudFront, consulte [Configurações de comportamento de cache](#) no Guia do CloudFront desenvolvedor.

3. Implante nós de borda da CDN próximos às populações de seus visualizadores. Trabalhe com seu provedor de CDN para:

- Identifique as localizações ideais dos nós periféricos com base nos dados demográficos dos espectadores

- Garanta capacidade suficiente em cada região
- Monitore o desempenho de ponta e ajuste conforme necessário

Para obter orientação de implementação, consulte os [CloudFront pontos](#) de presença para identificar os pontos de presença disponíveis para as regiões de seu público.

4. Para públicos globais, considere implementar uma estratégia de várias CDN. Essa abordagem:

- Usa vários provedores de CDN para melhorar a confiabilidade
- Encaminha os espectadores para a CDN com melhor desempenho de sua localização
- Fornece opções de failover durante interrupções do CDN
- Pode otimizar custos aproveitando diferentes modelos de preços

Para obter detalhes sobre a implementação, consulte [Estratégias de várias CDN](#) no blog AWS Networking & Content Delivery.

Personalize o planejamento para CDN e integrações MediaTailor

AWS Elemental MediaTailor Fluxos de trabalho diferentes têm requisitos exclusivos que afetam o planejamento da rede de distribuição de conteúdo (CDN). Para obter orientações específicas sobre o fluxo de trabalho, consulte Como [trabalhar](#) com configurações.

Ajuste seu plano de capacidade com base em seu MediaTailor fluxo de trabalho específico:

Para fluxos de MediaTailor trabalho de inserção de anúncios

1. Configure sua CDN para lidar com manifestos personalizados com zero armazenamento em cache. Isso garante que cada espectador receba anúncios exclusivos e direcionados. Os anúncios são baseados em seu perfil e contexto de visualização.
2. Dimensione seu servidor de decisão de anúncios (ADS) para lidar com volumes de solicitações de pico. Para obter orientação sobre a configuração do ADS, consulte [Inserção de anúncios com MediaTailor](#). Considere:
 - Requisitos de tempo de resposta para seu caso de uso
 - Capacidade esperada de visualização simultânea
 - Requisitos de redundância e failover
 - necessidades de distribuição geográfica

3. Implemente o colapso de solicitações no nível da CDN para lidar com solicitações sincronizadas de interrupção de anúncios. O colapso da solicitação combina várias solicitações idênticas em uma única solicitação de origem. Isso é crucial durante:

- Eventos esportivos ao vivo em que muitos espectadores acessam os intervalos publicitários simultaneamente
- Estreias de programas de TV populares com intervalos comerciais sincronizados
- Notícias de última hora que atraem picos repentinos de espectadores

Para obter detalhes de implementação, consulte [Failover do Origin](#) para configurar o tratamento de solicitações durante picos de carga.

Para fluxos de trabalho de montagem de MediaTailor canais

1. Calcule os requisitos de capacidade com base no número de canais e suas taxas de bits.

Para obter orientação sobre o planejamento da capacidade de montagem de canais, consulte [Montagem de canais em MediaTailor](#). Considere:

- Número total de canais
 - Requisitos de taxa de bits por canal
 - Carga de visualização simultânea esperada
 - necessidades de distribuição geográfica
2. Configure sua CDN para lidar com padrões de tráfego previsíveis com base em programações publicadas. Normalmente, a montagem de canais tem padrões mais previsíveis do que a inserção de anúncios porque:
- Os horários de programação são conhecidos com antecedência
 - O comportamento do espectador segue padrões estabelecidos
 - O conteúdo não muda dinamicamente por espectador
3. Certifique-se de que sua origem tenha largura de banda suficiente para manter a saída de canal consistente. Implemente:
- Servidores de origem redundantes para canais de alta disponibilidade
 - Failover automatizado entre as origens primária e de backup
 - Monitoramento para detectar problemas de desempenho de origem

Para obter orientação de implementação, consulte [Configurando a redundância de origem](#) para criar uma infraestrutura de origem resiliente.

Para fluxos de MediaTailor trabalho combinados

1. Dimensione sua infraestrutura para lidar com os padrões de tráfego combinados de ambos os serviços. Para obter orientação sobre fluxos de trabalho combinados, consulte [Utilizar AWS Elemental MediaTailor para inserir anúncios](#). Considere:
 - Requisitos básicos de montagem de canais
 - Requisitos de sobrecarga de inserção de anúncios
 - Padrões de pico de tráfego
 - Necessidades de redundância
2. Configure comportamentos CDN separados para entrega linear de conteúdo e inserção dinâmica de anúncios. Essa separação permite que você:
 - Otimize as políticas de armazenamento em cache para cada tipo de conteúdo de forma independente
 - Encaminhe as solicitações para as origens apropriadas com base no tipo de conteúdo
 - Monitore as métricas de desempenho separadamente para cada fluxo de trabalho
3. Configure o roteamento adequado entre a borda e CDNs a origem para manter o desempenho ideal. Considere usar:
 - Caminhos de origem diferentes para segmentos de conteúdo (/content/*) e segmentos de anúncios (/ads/*)
 - Comportamentos de cache separados para manifestos e segmentos
 - Roteamento geográfico para otimizar a latência para diferentes regiões

Para obter detalhes de implementação, consulte [Configuração de comportamentos de cache](#) para configurar regras de roteamento e armazenamento em cache com base em caminhos.

Configure o monitoramento e o escalonamento para CDN e integrações MediaTailor

Estratégias eficazes de monitoramento e escalonamento são cruciais para manter o desempenho ideal e a experiência do espectador com a integração da sua rede de distribuição de AWS Elemental MediaTailor conteúdo (CDN). Implemente essas abordagens para garantir que sua integração de CDN tenha um desempenho confiável em grande escala.

Implemente essas estratégias de monitoramento e escalonamento que seguem:

Configure o monitoramento e o escalonamento

1. Configure o monitoramento para essas métricas principais. Para obter orientação sobre os valores-alvo apropriados, consulte [Monitoramento MediaTailor com a Amazon CloudWatch](#) e consulte as melhores práticas do seu provedor de CDN:
 - Taxas de acerto do cache CDN (estabeleça métricas e metas básicas com base no seu tipo de conteúdo e padrões de entrega)
 - Volumes de solicitação de origem (monitore padrões durante a operação normal para estabelecer linhas de base para detecção de anomalias)
 - Taxas de erro por tipo de erro (defina limites com base em seus objetivos de nível de serviço e nas MediaTailor melhores práticas)
 - Tempos de resposta (defina metas de latência apropriadas com base nos requisitos de experiência do espectador e na distribuição geográfica)

Para obter instruções detalhadas de implementação, consulte [Criação de CloudWatch painéis](#) para visualizar suas MediaTailor métricas e da CDN juntas.

2. Configure alertas para padrões de tráfego inesperados ou degradação do desempenho. Configure limites com base em suas métricas básicas e objetivos de nível de serviço. Para obter orientação sobre como configurar alertas, consulte [Criação de CloudWatch alarmes na Amazon](#). Considere monitorar:
 - Desvios significativos das taxas de acerto do cache da linha de base (normalmente alertam quando estão abaixo de 85 a 90%)
 - Aumentos repentinos no volume de solicitações de origem (alerta sobre um aumento de 30% ou mais em relação à linha de base)
 - Picos de taxa de erro que excedem seus limites definidos (normalmente 1-2% para erros 4xx, 0,5% para erros 5xx)
 - Degradação do tempo de resposta além dos níveis aceitáveis (normalmente >500 ms para manifestos, >200 ms para segmentos)

Para exemplos de implementação, consulte [CloudWatch conceitos](#) para criar painéis e alertas de monitoramento eficazes.

3. Crie planos de escalabilidade para eventos previsíveis de alto tráfego. Seus planos devem incluir esses elementos-chave:
 - Aumento da capacidade pré-evento (24 a 48 horas antes do início do evento)
 - Cronogramas de aumento gradual de espectadores (normalmente de 10 a 20% do público esperado por intervalo de 5 minutos)

- Distribuição da capacidade regional com base no público (aloque a capacidade proporcionalmente à audiência regional esperada)
- Procedimentos de escalonamento pós-evento (mantenha a capacidade máxima por 30 a 60 minutos após a conclusão do evento)

Para obter orientações de implementação sobre escalabilidade para eventos de alto tráfego, consulte [Como configurar um fluxo de trabalho end-to-end ao vivo resiliente no AWS blog de mídia](#).

4. Implemente medidas de failover e redundância para fluxos críticos, incluindo:

- Implantações de CDN em várias regiões (pelo menos duas regiões para conteúdo crítico)
- Servidores de origem de backup (configurados com verificações de saúde automatizadas a cada 30 segundos)
- Acionadores de failover automatizados com base em verificações de integridade (normalmente após 2 a 3 verificações malsucedidas)
- Procedimentos de recuperação para diferentes cenários de falha (documentados com metas específicas de tempo de resposta)

Para obter etapas detalhadas de implementação, consulte [Otimizando a alta disponibilidade com failover de CloudFront origem](#).

Otimize os custos de CDN e integrações MediaTailor

Os custos da rede de entrega de conteúdo (CDN) podem variar significativamente com base nos padrões de tráfego, distribuição geográfica e uso de recursos com AWS Elemental MediaTailor. Para obter informações atuais sobre a definição de preço, consulte [Preços do CloudFront](#). Você também pode consultar a documentação do seu provedor de CDN.

Equilibre o desempenho com a eficiência de custos usando essas estratégias:

1. Analise seus padrões de tráfego de CDN para selecionar o nível de preços mais econômico. Analise os seguintes fatores com seu provedor de CDN:
 - Volumes de transferência de dados por região e período
 - Padrões de solicitação para manifestos e segmentos
 - Requisitos de distribuição geográfica
 - Padrões de uso de pico versus média

Para obter ajuda com a análise de custos, use a [Calculadora de AWS preços](#) para estimar seus CloudFront custos com base em seus padrões de uso específicos.

2. Para cargas de trabalho previsíveis, avalie os contratos de capacidade reservada com seu provedor de CDN. Esses acordos podem oferecer benefícios como:

- Tarifas com desconto para volumes de uso comprometidos
- Custos mensais previsíveis para orçamentação
- Suporte prioritário e alocação de capacidade

Consulte seu provedor de CDN para determinar se a capacidade reservada é apropriada para seus padrões de uso. Para ver CloudFront, consulte os [recursos CloudFront premium](#). Isso fornece informações sobre as opções de capacidade reservada.

3. Otimize os custos de saída equilibrando o tráfego entre MediaTailor e seu provedor de CDN. As estratégias incluem:

- Maximize as taxas de acerto do cache para reduzir as solicitações de origem
- Use o escudo de origem para consolidar solicitações
- Implemente a compactação para reduzir os volumes de transferência de dados
- Escolha regiões CDN que se alinhem aos AWS preços da região

Para obter orientações de implementação sobre compactação, consulte Como [servir arquivos compactados](#) no Guia do CloudFront desenvolvedor.

4. Implemente estratégias de armazenamento em cache apropriadas para diferentes tipos de conteúdo para reduzir as solicitações de origem. Para obter orientação sobre otimização do cache, consulte [Melhorando as taxas de acertos do cache](#). Concentre-se em:

- Segmentos de conteúdo (você pode armazená-los em cache por longos períodos)
- Segmentos de anúncios (normalmente reutilizados entre os espectadores)
- Ativos estáticos, como arquivos e imagens do player

As taxas de acerto de cache aprimoradas reduzem significativamente os custos de origem. Trabalhe com seu provedor de CDN para otimizar as configurações de cache para seus padrões de conteúdo específicos. Para obter etapas detalhadas de implementação, consulte [Configuração de comportamentos de cache](#).

Teste sua implementação para CDN e MediaTailor integrações

Garanta experiências confiáveis para os espectadores testando minuciosamente sua integração com a rede de distribuição de AWS Elemental MediaTailor conteúdo (CDN) antes da implantação na produção. Os testes adequados ajudam a identificar e resolver problemas antes que eles afetem seu público. Para obter orientação sobre metodologias de teste, consulte [Testar CloudFront distribuições](#). Você também pode consultar a documentação de teste do seu provedor de CDN.

Siga estas etapas de teste que validam sua integração com a CDN:

1. Crie um ambiente de teste que espelhe sua configuração de produção. Inclusão:

- Configurações de CDN e comportamentos de cache idênticos
- Inclua conteúdo representativo com várias taxas de bits e formatos
- Configure o servidor de decisão de anúncios de teste com exemplos de respostas de anúncios
- Definir configurações de monitoramento e alerta

Para obter orientação de step-by-step implementação, consulte [Criação de uma distribuição temporária](#) no Guia do CloudFront desenvolvedor.

2. Realize testes de carga para verificar suas estimativas de capacidade. Para obter orientação sobre testes de carga, consulte [Monitoramento MediaPackage](#). Os cenários de teste devem incluir:

- Aumento gradual da contagem de espectadores (normalmente de 10 a 20% do pico esperado a cada 5 minutos)
- Picos repentinos de tráfego com base nas cargas de pico esperadas (simule 50% do pico de público participando em 60 segundos)
- Períodos de pico de carga estendidos (mantenha o pico de carga por pelo menos 30 a 60 minutos)
- Distribuição geográfica correspondente ao seu público (distribua o tráfego de teste de acordo com a localização esperada dos espectadores)

Valide se os tempos de resposta permanecem abaixo dos limites desejados. Normalmente, isso significa menos de 500 ms para manifestos e menos de 200 ms para segmentos. As taxas de erro devem permanecer abaixo de 1%. Para obter detalhes de implementação sobre ferramentas e metodologias de teste de carga, consulte [Teste de carga com CloudFront](#) no blog AWS Networking & Content Delivery.

3. Teste cenários de failover para garantir a confiabilidade. Simule:

- Falhas no servidor de origem (cenários de interrupção completa e degradação parcial)
- Interrupções na localização periférica da CDN (teste com roteamento de tráfego para locais de backup)
- Indisponibilidade do servidor de decisão de anúncios (teste com tempos limite de 5 a 10 segundos)
- Problemas de conectividade de rede (simule perda de pacotes e aumentos de latência)

Trabalhe com seu provedor de CDN para estabelecer metas de tempo de resposta de failover adequadas para seu caso de uso. Normalmente, isso significa menos de 3 segundos para a conclusão do failover. Para obter orientações de implementação sobre testes de failover, consulte [Origin failover](#) no Guia do CloudFront desenvolvedor.

4. Para grandes eventos, implemente estratégias de aceleração gradual com base nas diretrizes de [teste de AWS carga](#):

- Horários de início escalonados para diferentes segmentos de público. Por exemplo, permita primeiro assinantes premium com intervalos de 15 minutos entre grupos de público.
- Use técnicas de pré-aquecimento para aumentar gradualmente a carga. O pré-aquecimento envolve:
 - Preencha caches de CDN com conteúdo popular de 2 a 4 horas antes do evento
 - Aumente gradualmente o tráfego sintético para 20-30% do pico esperado para aquecer os sistemas
 - Teste todos os componentes sob condições de carga realistas com conteúdo real
- Monitore o desempenho do sistema durante todo o período de inicialização, rastreando:
 - Taxas de acerto e tempos de resposta do cache (meta > 90% de taxa de acerto, resposta de <500 ms)
 - Taxas de erro e carga de origem (mantenha taxas de erro < 1%, CPU de origem < 70%)
 - Taxas de sucesso da personalização de anúncios (meta > 98% de personalização bem-sucedida)
 - Métricas de experiência do espectador (meta de <2 segundos de tempo de inicialização, < 0,5% de rebuffer)
- Tenha um plano de contingência para picos de tráfego inesperados. Seu plano deve incluir esses componentes essenciais:
 - Procedimentos de escalonamento de capacidade de emergência com etapas documentadas para aumentar a capacidade em 50 a 100% em 15 minutos

- Protocolos de ativação de CDN de backup com capacidade de transferir de 20 a 50% do tráfego para CDN secundária
- Recuo simplificado de inserção de anúncios para reduzir os parâmetros de segmentação de 10 ou mais para 3 a 5 parâmetros essenciais
- Planos de comunicação para partes interessadas com modelos de notificação e listas de contatos predefinidos

Depois de concluir o teste, continue com a [implementação da integração com a CDN](#) para as etapas de implantação da produção.

Solucione problemas comuns de CDN e integrações MediaTailor

Enfrente os desafios comuns de integração da rede de distribuição de conteúdo (CDN) AWS Elemental MediaTailor antes que eles afetem seus espectadores. Esta seção ajuda você a identificar e resolver problemas típicos que ocorrem durante a integração do CDN com o AWS Elemental MediaTailor

Para obter uma orientação abrangente sobre solução de problemas, consulte [Solução de problemas MediaTailor e solução de problemas de CloudFront distribuições](#).

Resolvendo problemas de entrega de MediaTailor manifestos

Se os espectadores tiverem problemas de reprodução ou virem anúncios incorretos, verifique estes problemas comuns relacionados ao manifesto:

- Configurações de cache incorretas: se sua CDN armazenar em cache manifestos personalizados, os espectadores poderão ver anúncios destinados a outros usuários.

Solução: configure sua CDN com um TTL de cache de 0 para solicitações de manifesto. MediaTailor

- Falhas na solicitação de origem: se sua CDN não puder ser acessada MediaTailor, as solicitações de manifesto falharão.

Solução: verifique a conectividade de rede entre seu CDN e MediaTailor. Verifique se seu CDN encaminha corretamente os cabeçalhos corretos.

- Problemas de parâmetros de sessão: parâmetros de sessão ausentes ou incorretos podem causar falhas na personalização.

Solução: certifique-se de que seu player esteja anexando corretamente todos os parâmetros de sessão necessários às solicitações do manifesto.

Corrigindo problemas de entrega do MediaTailor segmento

Se o conteúdo ou os segmentos de anúncios não estiverem sendo carregados corretamente, investigue estes problemas comuns:

- **Regravação do caminho do segmento:** a configuração incorreta do CDN pode reescrever o segmento de forma inadequada. URLs

Solução: verifique se sua CDN lida corretamente com o segmento URLs. Certifique-se de que ele não modifique os caminhos de uma forma que quebre as referências.

- **Configuração do CORS:** cabeçalhos CORS ausentes ou incorretos podem impedir que os navegadores carreguem segmentos.

Solução: configure sua CDN para passar os cabeçalhos CORS apropriados para solicitações de segmentos.

- **Tempestades de falhas de cache:** durante eventos de alto tráfego, várias falhas de cache podem sobrecarregar os servidores de origem.

Solução: implemente recursos de redução de solicitações e proteção de origem para reduzir a carga de origem durante picos de tráfego.

Resolvendo problemas MediaTailor de desempenho da CDN

Se os espectadores tiverem problemas de buffer ou carregamento lento, verifique estes problemas relacionados ao desempenho:

- **Baixa taxa de acerto do cache:** se sua CDN frequentemente solicita conteúdo da origem, o desempenho será prejudicado.

Solução: analise as taxas de acertos do cache por tipo de conteúdo e ajuste as configurações de TTL para melhorar a eficiência do armazenamento em cache.

- **Distribuição geográfica:** os espectadores distantes dos pontos de presença da CDN podem experimentar maior latência.

Solução: revise sua distribuição de pontos de presença na CDN. Aumente a capacidade em regiões com alta concentração de espectadores.

- Limitações de capacidade de origem: se seus servidores de origem estiverem sobrecarregados, os tempos de resposta aumentarão.

Solução: Implemente a limitação da solicitação de origem. Você também pode aumentar a capacidade de origem ou melhorar o armazenamento em cache para reduzir a carga de origem.

Para obter assistência adicional para solução de problemas, consulte [Solução de problemas MediaTailor](#) ou entre em contato com o AWS Support. Para obter orientações de implementação sobre como resolver problemas comuns de CDN, consulte [Depurando sua rede de distribuição de conteúdo no blog de mídia](#). AWS

Configure a integração da CDN com MediaTailor

Esta seção fornece orientação sobre a integração AWS Elemental MediaTailor com uma rede de distribuição de conteúdo (CDN).

A integração efetiva da CDN com MediaTailor é essencial para oferecer experiências de streaming de alta qualidade com anúncios personalizados em grande escala. Este guia orienta você pelo processo completo de configuração, configuração e otimização da integração com a CDN.

Para obter informações adicionais, consulte os links a seguir:

- Para obter informações sobre como transmitir parâmetros de consulta CDNs para autorização e segmentação de anúncios, consulte [MediaTailor parâmetros de consulta do manifesto](#).
- Para roteamento avançado usando variáveis dinâmicas e aliases de configuração, consulte [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#).
- Para obter informações sobre a criação MediaTailor de configurações, consulte [Utilizar AWS Elemental MediaTailor para inserir anúncios](#).
- Para obter informações sobre como criar uma CloudFront distribuição, consulte [Criação de uma distribuição](#) no Guia do CloudFront desenvolvedor.

Para garantir clareza neste guia, a terminologia a seguir é usada para descrever diferentes tipos de manifestos:

- O HLS manifesta:
 - Playlist multivariante: o manifesto de nível superior que contém links para playlists de mídia
 - Playlist de mídia: o manifesto de segundo nível com links para segmentos de conteúdo
- O DASH manifesta:
 - MPD (Descrição da apresentação de mídia): o termo padrão para manifestos do DASH

A integração MediaTailor com uma CDN oferece os seguintes benefícios:

- Latência reduzida para espectadores
- Escalabilidade aprimorada para eventos de alto tráfego
- Confiabilidade aprimorada por meio de caminhos de entrega redundantes
- Custos otimizados reduzindo o tráfego de origem
- Melhor proteção contra ataques DDoS

Componentes e requisitos de integração de CDN

Uma integração bem-sucedida da CDN MediaTailor envolve a configuração dos seguintes componentes principais:

Comportamentos de roteamento de CDN

Regras que determinam como diferentes tipos de solicitações (manifestos, segmentos de conteúdo, segmentos de anúncios) são roteados por meio de sua CDN.

Mapeamento de CDN em MediaTailor

Configuração MediaTailor que garante que os manifestos façam referência ao seu domínio CDN em vez de referenciar diretamente os servidores de origem.

Configurações de segurança

Configurações que protegem seu conteúdo e sua infraestrutura, incluindo segurança de transporte, controle de acesso e monitoramento.

Testes e validação

Procedimentos para verificar se sua integração de CDN está funcionando corretamente antes da implantação na produção.

Pré-requisitos para integração com CDN

Antes de configurar sua integração com a CDN, verifique se você tem o seguinte:

1. Uma MediaTailor configuração com as seguintes configurações:

- Sua origem de conteúdo como fonte de conteúdo
- Seu ADS como servidor de decisão de anúncios

Você também precisa da origem e do ADS URLs nas etapas de integração do CDN.

2. Acesso à interface de configuração da sua CDN
3. Compreensão da terminologia específica da sua CDN para comportamentos, regras e configurações de cache
4. Conhecimento de sua estrutura de conteúdo, incluindo extensões de arquivo usadas para segmentos (como .ts, .mp4 ou .m4s)

Etapas de configuração da integração CDN

O processo de integração MediaTailor com uma CDN segue estas etapas de alto nível:

1. Configurar comportamentos de roteamento de CDN - Configure seu CDN para rotear diferentes tipos de solicitações de forma adequada.
2. Configure o mapeamento CDN em MediaTailor - Atualize sua MediaTailor configuração para usar seus nomes de domínio CDN.
3. Implemente as melhores práticas de segurança — defina as configurações de segurança para proteger seu conteúdo e sua infraestrutura.
4. Teste sua integração - Verifique se sua integração com a CDN está funcionando corretamente.

Cabeçalhos necessários para integração com MediaTailor CDN

MediaTailor Para funcionar corretamente com sua CDN, você deve configurar sua CDN para encaminhar cabeçalhos HTTP específicos. Esses cabeçalhos são essenciais para a funcionalidade adequada, incluindo compressão, detecção de dispositivos, personalização de anúncios e segmentação geográfica.

Configure sua CDN para encaminhar os seguintes cabeçalhos para: MediaTailor

Accept-Encoding

Objetivo: Necessário para a funcionalidade de compressão

Detalhes: Esse cabeçalho informa MediaTailor quais métodos de compactação o cliente suporta. MediaTailor usa essas informações para compactar manifestos quando possível, reduzindo o uso da largura de banda e melhorando o desempenho. Dispositivos antigos que não oferecem suporte à compactação não enviarão esse cabeçalho e MediaTailor retornarão manifestos não compactados.

User-Agent

Objetivo: necessário para detecção de dispositivos e personalização de anúncios

Detalhes: MediaTailor analisa o cabeçalho User-Agent para identificar o tipo de dispositivo cliente, o navegador e os recursos. Essas informações são usadas para segmentação de anúncios, otimizações específicas de dispositivos e garantia de compatibilidade com diferentes clientes de reprodução.

Host

Objetivo: Necessário para o roteamento adequado da solicitação

Detalhes: o Host cabeçalho garante que as solicitações sejam roteadas para o MediaTailor endpoint correto. Isso é particularmente importante em ambientes multilocatários e ao usar configurações de domínio personalizadas.

Muitos CDNs, incluindo a Amazon CloudFront, não encaminham o Host cabeçalho por padrão. Para CloudFront usuários: consulte [Configurando comportamentos de cache](#) para obter instruções de configuração.

X-Forwarded-For

Objetivo: Necessário para detecção de IP do cliente e segmentação geográfica

Detalhes: esse cabeçalho preserva o endereço IP original do cliente quando as solicitações passam pela sua CDN. MediaTailor usa essas informações para segmentação geográfica de anúncios, análises e conformidade com restrições de conteúdo regionais.

Important

Todos os quatro cabeçalhos são necessários para a MediaTailor funcionalidade completa. A falta de qualquer um desses cabeçalhos pode resultar na redução da funcionalidade, incluindo:

- Incapacidade de compactar manifestos (falta a codificação de aceitação)
- Segmentação deficiente de anúncios e problemas de compatibilidade de dispositivos (falta o User-Agent)
- Falhas no roteamento de solicitações (host ausente)
- Segmentação geográfica e análise imprecisas (ausentes) X-Forwarded-For

Para obter instruções de configuração específicas da CDN, consulte as seções de comportamento de roteamento e armazenamento em cache que fazem referência a essa lista de cabeçalhos.

Os tópicos a seguir fornecem instruções detalhadas para cada aspecto da integração com a CDN.

Tópicos

- [Configure comportamentos de roteamento de CDN para MediaTailor](#)
- [Configure o mapeamento de CDN em MediaTailor](#)
- [Melhores práticas de segurança de integração de CDN para MediaTailor](#)

Configure comportamentos de roteamento de CDN para MediaTailor

Esta seção explica como configurar sua rede de distribuição de conteúdo (CDN) para rotear diferentes tipos de solicitações de forma adequada para AWS Elemental MediaTailor integração. A configuração de roteamento adequada garante que as solicitações de manifesto, os segmentos de conteúdo e os segmentos de anúncios sejam tratados corretamente.

Configurar os comportamentos de roteamento da CDN é uma etapa crítica na criação de um pipeline eficiente de entrega de conteúdo. Ao configurar regras de roteamento específicas para diferentes tipos de conteúdo, você pode otimizar o armazenamento em cache, melhorar o desempenho de entrega e garantir que a inserção de anúncios personalizados funcione corretamente.

Para cenários avançados de roteamento usando variáveis dinâmicas e aliases de configuração, consulte. [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#) Para obter

informações sobre como preservar os parâmetros de consulta no roteamento CDN, consulte.

[MediaTailor parâmetros de consulta do manifesto](#)

 Important

A falha na inclusão de cabeçalhos CORS no objeto em cache ou na resposta da CDN aos visualizadores pode causar falhas na reprodução.

Configuração do comportamento de roteamento CDN

Configure sua CDN para rotear diferentes tipos de solicitações de forma adequada.

Roteamento de segmentos de conteúdo

O roteamento de segmentos de conteúdo direciona as solicitações de seus segmentos de conteúdo reais para o servidor de origem. Assim como o roteamento de segmentos de anúncios, o roteamento de segmentos de conteúdo também requer uma configuração adequada do CORS para garantir uma reprodução suave em players baseados na web.

Para obter orientações detalhadas de configuração, consulte o CloudFront exemplo em [Precedência 4: comportamento do caminho de origem do conteúdo](#). Este exemplo fornece configurações específicas que você deve seguir CloudFront ou adaptar para outras CDNs.

Os principais requisitos de configuração para roteamento de segmentos de conteúdo incluem:

- Use padrões de caminho que correspondam às extensões de arquivo do segmento de conteúdo (como *.ts*.mp4, ou *.m4s)
- Encaminhe solicitações para sua origem de conteúdo (como um bucket ou MediaPackage endpoint do Amazon S3)
- Para obter uma taxa de acerto de cache ideal, inclua somente parâmetros de sequência de caracteres de consulta que façam com que sua origem modifique a resposta na chave de cache e encaminhe a solicitação de origem
- Aplique uma política de cache apropriada com valores TTL maiores que 24 horas
- Inclua cabeçalhos de resposta do CORS para seus espectadores

Roteamento de segmentos de anúncios

O roteamento do segmento de anúncios é fundamental para oferecer anúncios personalizados aos espectadores. Ao configurar o roteamento do segmento de anúncios, você deve implementar o tratamento adequado do CORS (Cross-Origin Resource Sharing) para evitar problemas que possam causar falhas de reprodução em players baseados na web.

Para obter orientações detalhadas de configuração, consulte o CloudFront exemplo em [Precedência 0: comportamento do caminho dos segmentos de anúncios](#). Este exemplo fornece configurações específicas que você deve seguir CloudFront ou adaptar para outras CDNs.

Os principais requisitos de configuração para roteamento de segmentos de anúncios incluem:

- Use o padrão de caminho `/tm/*` especificamente para segmentos de MediaTailor anúncios
- Encaminhar solicitações para `segments.mediatailor.region.amazonaws.com`
- Para obter uma taxa de acerto de cache ideal, não inclua nenhum cabeçalho de solicitação do visualizador, cookies ou parâmetros de sequência de caracteres de consulta na chave de cache ou na solicitação de origem
- Aplique uma política de cache apropriada com valores TTL maiores que 24 horas
- Inclua cabeçalhos de resposta do CORS para seus espectadores

Roteamento de solicitações de manifesto

Para encaminhar solicitações de playlists multivariantes, playlists de mídia e MPD para MediaTailor, use as seguintes configurações gerais. Para CloudFront configuração, consulte [Configurando o comportamento do cache do manifesto](#).

1. Na sua interface de configuração de CDN, crie comportamentos para diferentes tipos de manifesto.
2. Defina padrões de caminho para combinar extensões multivariantes de arquivos de playlists e playlists de mídia (`*.m3u8` para HLS) e extensões de arquivo MPD (para DASH). `*.mpd`
3. Defina a configuração de origem em sua CDN para apontar para seu endpoint MediaTailor de configuração.
4. Para inserção de anúncios, desative o armazenamento em cache de playlists multivariantes personalizadas, playlists de mídia e MPDs. Como a inserção de anúncios fornece manifestos personalizados, sua CDN não deve armazená-los em cache. Se um dispositivo de reprodução diferente do pretendido receber uma playlist em cache ou MPD, isso poderá resultar em

problemas com a reprodução ou o rastreamento. Para obter orientações abrangentes sobre armazenamento em cache, incluindo recomendações de TTL para todos os tipos de conteúdo, consulte [Otimização de cache para CDN e integrações MediaTailor](#)

5. Configure o encaminhamento de cabeçalhos para todos os cabeçalhos. Para obter os requisitos mínimos, consulte [Cabeçalhos necessários para integração com MediaTailor CDN](#).
6. Ative o encaminhamento da sequência de caracteres de consulta para transmitir parâmetros de segmentação de anúncios.

HLS multivariant playlist

As solicitações de playlists multivariantes do HLS seguem estes formatos:

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

Exemplo:

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

HLS media playlist

As solicitações de playlist de mídia HLS seguem estes formatos:

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

As solicitações do jogador `https://CDN_Hostname/some/path/asset.m3u8` são roteadas para o MediaTailor caminho `https://mediataylor.us-west-2.amazonaws.com/v1/session/configuration/endpoint` com base na palavra-chave `*.m3u8` na solicitação.

Exemplo:

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

DASH MPD

As solicitações DASH MPD seguem estes formatos:

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

As solicitações do jogador `https://CDN_Hostname/some/path/asset.mpd` são roteadas para o MediaTailor caminho `https://mediatailor.us-west-2.amazonaws.com/v1/dash/configuration/endpoint` com base na palavra-chave `*.mpd` na solicitação.

Exemplo:

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

Melhores práticas de roteamento de CDN

Ao configurar os comportamentos de roteamento da CDN, siga estas melhores práticas para garantir desempenho e confiabilidade ideais:

Use padrões de caminho específicos

Crie padrões de caminho específicos que correspondam com precisão à sua estrutura de conteúdo para garantir o roteamento adequado.

Priorize a ordem do comportamento

Na maioria CDNs, os comportamentos são avaliados em ordem. Coloque comportamentos mais específicos antes dos mais gerais.

Teste padrões de comportamento

Verifique se seus padrões de caminho correspondem corretamente às solicitações esperadas antes da implantação na produção.

Documente sua configuração

Mantenha a documentação dos comportamentos de roteamento de sua CDN para facilitar a solução de problemas e futuras atualizações.

Próximas etapas

Depois de configurar seus comportamentos de roteamento de CDN, a próxima etapa é configurar o mapeamento de CDN em MediaTailor. Para obter instruções, consulte [Configure o mapeamento de CDN em MediaTailor](#).

Configure o mapeamento de CDN em MediaTailor

Esta seção explica como configurar AWS Elemental MediaTailor para usar seus nomes de domínio da rede de distribuição de conteúdo (CDN). Depois de configurar seus comportamentos de roteamento de CDN, você precisa atualizar sua MediaTailor configuração para garantir que os manifestos façam referência ao seu domínio de CDN em vez de referenciar diretamente os servidores de origem.

A configuração do mapeamento de CDN MediaTailor garante que todo o conteúdo e segmento de anúncios URLs em seus manifestos apontem para sua CDN em vez de diretamente para os servidores de origem. Essa etapa é essencial para criar uma cadeia de entrega de CDN completa e maximizar os benefícios de sua integração com a CDN.

Configuração de mapeamento CDN em MediaTailor

Depois de configurar seus comportamentos de roteamento CDN, configure MediaTailor para usar seus nomes de domínio CDN:

1. Abra o [console do MediaTailor](#).
2. Selecione a configuração que você deseja atualizar.
3. Nas configurações avançadas da seção de configuração da CDN, insira o nome do seu domínio da CDN no campo de prefixo do segmento de conteúdo da CDN.
4. Se você estiver usando um domínio CDN separado para segmentos de anúncios, insira-o no campo Prefixo do segmento de anúncios CDN.
5. Salve as alterações.

Essa configuração garante a geração de MediaTailor manifestos com URLs desse ponto para sua CDN em vez de diretamente para a origem ou o armazenamento do segmento de anúncios.

Entendendo o comportamento do URL base

MediaTailor determina a base URLs nos manifestos com base na configuração do prefixo CDN:

- Prefixo do segmento de anúncio CDN configurado: os segmentos de anúncios usam o prefixo CDN como URL base.
- Prefixo do segmento de anúncio CDN não configurado: os segmentos de anúncios usam a MediaTailor localização direta como URL base.

- Prefixo do segmento de conteúdo CDN configurado: os segmentos de conteúdo usam o prefixo CDN como URL base.
- Prefixo do segmento de conteúdo CDN não configurado: os segmentos de conteúdo fazem referência à origem do conteúdo original.

Manipulação de URL base do DASH

Para manifestos do DASH, MediaTailor gerencia BaseURL as configurações de forma diferente para segmentos de conteúdo e anúncios:

Segmentos de conteúdo:

- Com o prefixo do segmento de conteúdo CDN: MediaTailor define exatamente um BaseURL no MPD nível usando o prefixo especificado.
- Sem o prefixo do segmento de conteúdo CDN: MediaTailor preserva as BaseURL configurações existentes do manifesto de origem ou adiciona uma com base na MPD URL de origem, se não houver nenhuma.

Segmentos de anúncios:

- Com o prefixo do segmento de anúncio CDN: cada período do anúncio recebe exatamente um BaseURL usando o prefixo configurado.
- Sem o prefixo do segmento de anúncio CDN: cada período do anúncio tem exatamente um BaseURL apontando para o servidor de conteúdo do MediaTailor anúncio.

Exemplo Exemplo de mapeamento de CDN

Se sua origem de conteúdo for `http://origin.com/contentpath/` e seu prefixo de segmento de conteúdo CDN for `https://cdn.example.com/`, um segmento de conteúdo que normalmente seria referenciado como `http://origin.com/contentpath/subdir/content.ts` aparecerá no manifesto como `https://cdn.example.com/subdir/content.ts`

Considerações importantes

Ao configurar o mapeamento de CDN MediaTailor, tenha em mente as seguintes considerações importantes:

Use HTTPS para prefixos CDN

Sempre use HTTPS URLs para seus prefixos de CDN para garantir a entrega segura de conteúdo.

Combine os caminhos de comportamento do CDN

Certifique-se de que os prefixos de CDN que você configura MediaTailor correspondam aos padrões de caminho que você configurou em seus comportamentos de CDN.

Considere a região CDNs

Se você estiver usando domínios CDN diferentes para regiões diferentes, precisará criar MediaTailor configurações separadas para cada região.

Confirmar a propriedade de um domínio

Certifique-se de ter controle sobre os domínios CDN nos quais você configura. MediaTailor

Verifique a configuração de mapeamento do CDN

Depois de configurar o mapeamento de CDN MediaTailor, verifique se sua configuração está funcionando corretamente:

1. Solicite um manifesto por meio de sua CDN.
2. Examine o conteúdo do manifesto para verificar se o segmento faz URLs referência ao seu domínio CDN.
3. Verifique esse segmento de conteúdo URLs no ponto de manifesto para seu domínio CDN.
4. Verifique se o segmento de anúncio URLs no manifesto aponta para seu domínio CDN.

Para obter procedimentos abrangentes de teste e validação, consulte [Teste e validação para CDN e integrações MediaTailor](#).

Próximas etapas

Depois de configurar o mapeamento de CDN MediaTailor, a próxima etapa é implementar as melhores práticas de segurança para sua integração de CDN. Para obter instruções, consulte [Melhores práticas de segurança de integração de CDN para MediaTailor](#).

Melhores práticas de segurança de integração de CDN para MediaTailor

Esta seção explica como implementar as melhores práticas de segurança para sua integração com a rede de distribuição de AWS Elemental MediaTailor conteúdo (CDN). A configuração de segurança adequada ajuda a proteger seu conteúdo e sua infraestrutura contra acesso não autorizado e possíveis ameaças.

Proteger sua integração com a CDN é crucial para proteger seu conteúdo, impedir o acesso não autorizado e garantir a conformidade com os requisitos de segurança. A implementação dessas melhores práticas ajuda a criar uma postura de segurança robusta para seu fluxo de trabalho de streaming.

Configuração de segurança CDN

Antes de definir as configurações de roteamento e cache da CDN, implemente estas melhores práticas de segurança para proteger seu conteúdo e sua infraestrutura:

Segurança de transporte

MediaTailor usa apenas HTTPS para todas as comunicações e não permite conexões HTTP. Para garantir a comunicação segura entre todos os componentes do seu fluxo de trabalho de streaming, conclua as seguintes etapas:

1. Configure HTTPS para toda a comunicação entre sua CDN e MediaTailor endpoints. MediaTailor requer HTTPS e não aceitará conexões HTTP.
2. Use o TLS 1.2 ou posterior para todas as conexões HTTPS.
3. Configure sua CDN para impor conexões somente HTTPS dos visualizadores.

Controle de acesso

Para proteger seu conteúdo e sua infraestrutura contra acesso não autorizado, implemente os seguintes controles de acesso:

1. Defina as configurações de restrição geográfica se precisar limitar o acesso ao conteúdo a regiões específicas.
2. Implemente cookies assinados URLs ou assinados para conteúdo que exija autenticação do visualizador.
3. Configure listas de permissões baseadas em IP para acesso administrativo à configuração da CDN.

Monitoramento de segurança

Para detectar e responder aos eventos de segurança de forma eficaz, implemente as seguintes práticas de monitoramento:

1. Ative o registro de acesso para sua distribuição CDN.
2. Configure alertas para padrões de tráfego incomuns ou tentativas de acesso.
3. Revise regularmente as configurações de segurança e atualize conforme necessário.

Recursos avançados de segurança

Os seguintes recursos avançados de segurança fornecem proteção aprimorada para sua integração com a CDN:

Web Application Firewall (WAF)

Se sua CDN suportar a funcionalidade WAF, configure-a para se proteger contra vulnerabilidades e ataques comuns na web.

DDoSProteção S

Ative DDoS os recursos de proteção S fornecidos pela sua CDN para mitigar ataques distribuídos denial-of-service.

Controle de acesso do Origin

Configure seus servidores de origem para aceitar solicitações somente da sua CDN para evitar tentativas de acesso direto.

Criptografia de conteúdo

Para conteúdo altamente confidencial, considere implementar mecanismos adicionais de criptografia além da segurança do transporte.

Próximas etapas

Depois de implementar as melhores práticas de segurança, a próxima etapa é testar e solucionar problemas de sua integração com a CDN. Consulte [Solucionar problemas de integração com a CDN](#) para obter instruções abrangentes sobre testes e solução de problemas.

Configure o SSAI com uma CDN para publicidade em vídeo personalizada

Esta seção fornece uma orientação abrangente para integrar a inserção de anúncios do AWS Elemental MediaTailor lado do servidor (SSAI) com uma rede de entrega de conteúdo (CDN). Siga estas etapas para instalar, configurar e otimizar sua integração SSAI CDN.

A inserção de anúncios no lado do servidor (SSAI) é uma tecnologia que insere perfeitamente anúncios personalizados em streams de vídeo no nível do servidor, e não no nível do cliente. Quando combinado com uma CDN, você cria uma solução robusta e escalável para oferecer publicidade personalizada para públicos globais com latência mínima.

Neste tópico, usamos o termo manifestos para nos referirmos coletivamente a playlists multivariantes, playlists de mídia e. MPDs

O que você precisará

Antes de configurar a inserção de MediaTailor anúncios com uma CDN, reúna os seguintes recursos necessários:

AWS conta e permissões

- Uma AWS conta com permissões apropriadas para criar e gerenciar MediaTailor recursos

- Permissões do IAM para MediaTailor, CloudFront (se estiver usando) e serviços relacionados

- Para obter os requisitos de permissão detalhados, consulte [Segurança em AWS Elemental MediaTailor](#).

Serviços necessários

- AWS Elemental MediaTailor configurado e em execução

- Uma conta de rede de distribuição de conteúdo (CDN) (Amazon CloudFront ou CDN de terceiros)

- Servidor de origem para seu conteúdo (HLS ou DASH)

- Servidor de decisão de anúncios (ADS) compatível com VAST ou VMAP

Requisitos de conteúdo

- Conteúdo devidamente codificado e empacotado no formato HLS ou DASH

Marcadores de quebra de anúncio em seu conteúdo (para VOD) ou marcadores SCTE-35 (para transmissão ao vivo)

Antes de começar

Conclua essas tarefas de configuração antes de implementar a inserção de MediaTailor anúncios com uma CDN:

1. Configure a conectividade de rede entre seu CDN e MediaTailor servidores de origem
2. Configure HTTPS para entrega segura de conteúdo
3. Defina as configurações de DNS para seu domínio CDN
4. Para MediaTailor configuração básica, conclua as etapas em [Configurar Introdução à inserção MediaTailor de anúncios](#) e.

Pré-requisitos de conhecimento

Para implementar essa solução com sucesso, você deve ter:

- Compreensão dos protocolos de streaming (HLS/DASH)
- Conhecimento básico dos princípios de configuração da CDN
- Familiaridade com os conceitos de inserção de anúncios

Benefícios da integração com a CDN

A integração da SSAI com uma CDN oferece os seguintes benefícios principais:

Experiência aprimorada do espectador

A configuração adequada CDNs reduz o buffer, os tempos de inicialização e os erros de reprodução durante as transições de anúncios. Isso resulta em maior engajamento e satisfação do espectador.

Redução de custos

Estratégias eficientes de armazenamento em cache minimizam as solicitações de origem. Isso reduz os custos de transferência de dados e a carga do servidor de origem, particularmente importante para conteúdo de alto volume suportado por anúncios.

Escalabilidade

As configurações otimizadas de CDN lidam com picos de tráfego durante eventos populares sem degradar o desempenho, garantindo que seus anúncios personalizados sejam entregues mesmo nos horários de pico de visualização.

Alcance global

Configurado corretamente, CDNs forneça conteúdo com baixa latência para espectadores em todo o mundo, independentemente de sua localização, expandindo seu público potencial.

Transições de anúncios perfeitas

A configuração otimizada de CDN garante transições suaves entre conteúdo e anúncios, criando uma experiência de visualização com qualidade de transmissão.

Os tópicos a seguir fornecem orientação abrangente sobre a configuração MediaTailor com uma CDN para um desempenho ideal.

Tópicos

- [Entenda a arquitetura de inserção de anúncios para CDN e integrações MediaTailor](#)
- [Configure o MediaTailor SSAI básico com uma CDN para otimizar a entrega de anúncios](#)
- [Integre o MediaTailor SSAI com a montagem de canais para canais lineares monetizados](#)
- [Otimize o desempenho para CDN e integrações MediaTailor](#)
- [Monitore as operações de CDN e MediaTailor integrações](#)
- [Solucione problemas de MediaTailor SSAI com a entrega ininterrupta CDNs de anúncios](#)

Entenda a arquitetura de inserção de anúncios para CDN e integrações MediaTailor

Esta seção explica os conceitos e a arquitetura da inserção de anúncios no lado do servidor (SSAI) com redes de entrega de conteúdo () para. CDNs AWS Elemental MediaTailor Você aprenderá como a inserção dinâmica de anúncios e a manipulação de manifestos funcionam juntas para permitir uma monetização eficaz de vídeos.

A inserção de anúncios no lado do servidor (SSAI) permite que você: MediaTailor

- Insira publicidade personalizada em seus streams de vídeo em pontos de interrupção de anúncios definidos

- Segmente anúncios com precisão com base nos dados do espectador
- Elimine a necessidade de tecnologia de inserção de anúncios do lado do cliente

Quando combinado com uma CDN, você pode fornecer esses streams personalizados aos espectadores com melhor desempenho e escalabilidade, aprimorando sua estratégia de monetização de vídeos.

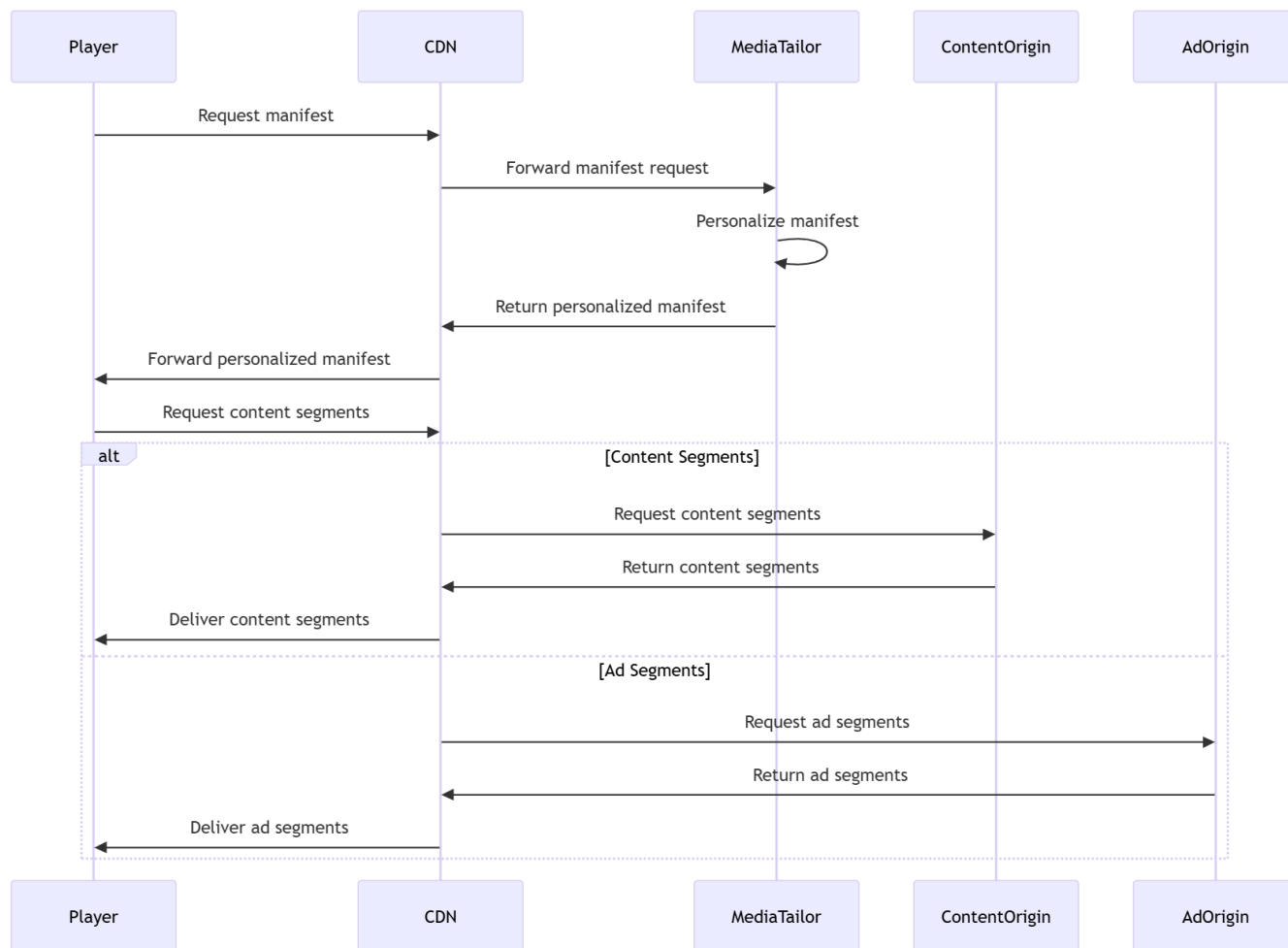
A arquitetura recomendada para inserção de anúncios com uma CDN posiciona a CDN entre os visualizadores e a inserção de anúncios, com a inserção de anúncios acessando o conteúdo diretamente de sua origem. Essa arquitetura oferece os seguintes benefícios tanto para a entrega de conteúdo quanto para a monetização de vídeos:

- Armazenamento efetivo de conteúdo e segmentos de anúncios
- Carga de solicitações reduzida em MediaTailor
- Velocidade de entrega aprimorada para os espectadores
- Gerenciamento simplificado de URL
- Entrega consistente de publicidade personalizada em todos os dispositivos

Nessa arquitetura recomendada:

1. Os espectadores solicitam manifestos da CDN
2. CDN encaminha solicitações para inserção de anúncios
3. A inserção de anúncios solicita manifestos de conteúdo da origem
4. A inserção de anúncios solicita anúncios do servidor de decisão de anúncios (ADS)
5. A inserção de anúncios personaliza os manifestos substituindo os marcadores de anúncios (do manifesto de origem) por URLs desse ponto em segmentos de anúncios segmentados para o espectador específico (do ADS)
6. A inserção de anúncios retorna os manifestos personalizados contendo segmentos de anúncios URLs para a CDN, que os encaminha para os espectadores
7. Os espectadores solicitam segmentos por meio da CDN
8. A CDN encaminha as solicitações de segmentos com base no tipo de segmento:
 - As solicitações de segmentos de conteúdo vão para a origem do conteúdo
 - As solicitações de segmentos de anúncios vão para MediaTailor

Essa arquitetura garante um desempenho ideal enquanto mantém os benefícios de segurança e flexibilidade do uso de uma CDN.



Note

Esse fluxo varia um pouco entre VOD e conteúdo ao vivo. Para VOD, os manifestos podem ser armazenados em cache por mais tempo, enquanto o conteúdo ao vivo exige atualizações mais frequentes dos manifestos para manter a continuidade do stream.

A principal diferença entre VOD e cache de conteúdo ao vivo:

Conteúdo VOD

Defina valores de TTL mais longos (minutos a horas) para manifestos porque eles não mudam com frequência

Conteúdo ao vivo

Defina valores de TTL mais curtos (segundos) para manifestos para garantir que os espectadores recebam os segmentos de transmissão mais atuais

Não recomendamos que você posicione uma CDN entre a origem do seu conteúdo e AWS Elemental MediaTailor. Fazer isso pode introduzir vários desafios técnicos:

Colisões de chaves de cache

Configure sua CDN para lidar adequadamente com os parâmetros de consulta. Isso evita o recebimento MediaTailor de manifestos incorretos ao solicitar o mesmo manifesto com parâmetros de consulta diferentes.

Problemas de compressão do Gzip

Se você tiver erros de análise de manifestos, certifique-se de que sua CDN entregue manifestos formatados corretamente para MediaTailor. Alguns CDNs podem fornecer cargas gzip corrompidas que podem causar falhas na análise. Se isso ocorrer, talvez seja necessário desativar a compactação entre sua CDN e, ao MediaTailor mesmo tempo, manter a compactação para reduzir custos em outras partes do fluxo de trabalho.

Frescura manifesta

Para transmissões ao vivo, configure sua CDN para entregar os manifestos atuais para MediaTailor. Isso evita problemas de sincronização entre conteúdo e anúncios.

Otimização de desempenho

Minimize os saltos de rede e possíveis perdas de cache para reduzir os tempos de inicialização da reprodução.

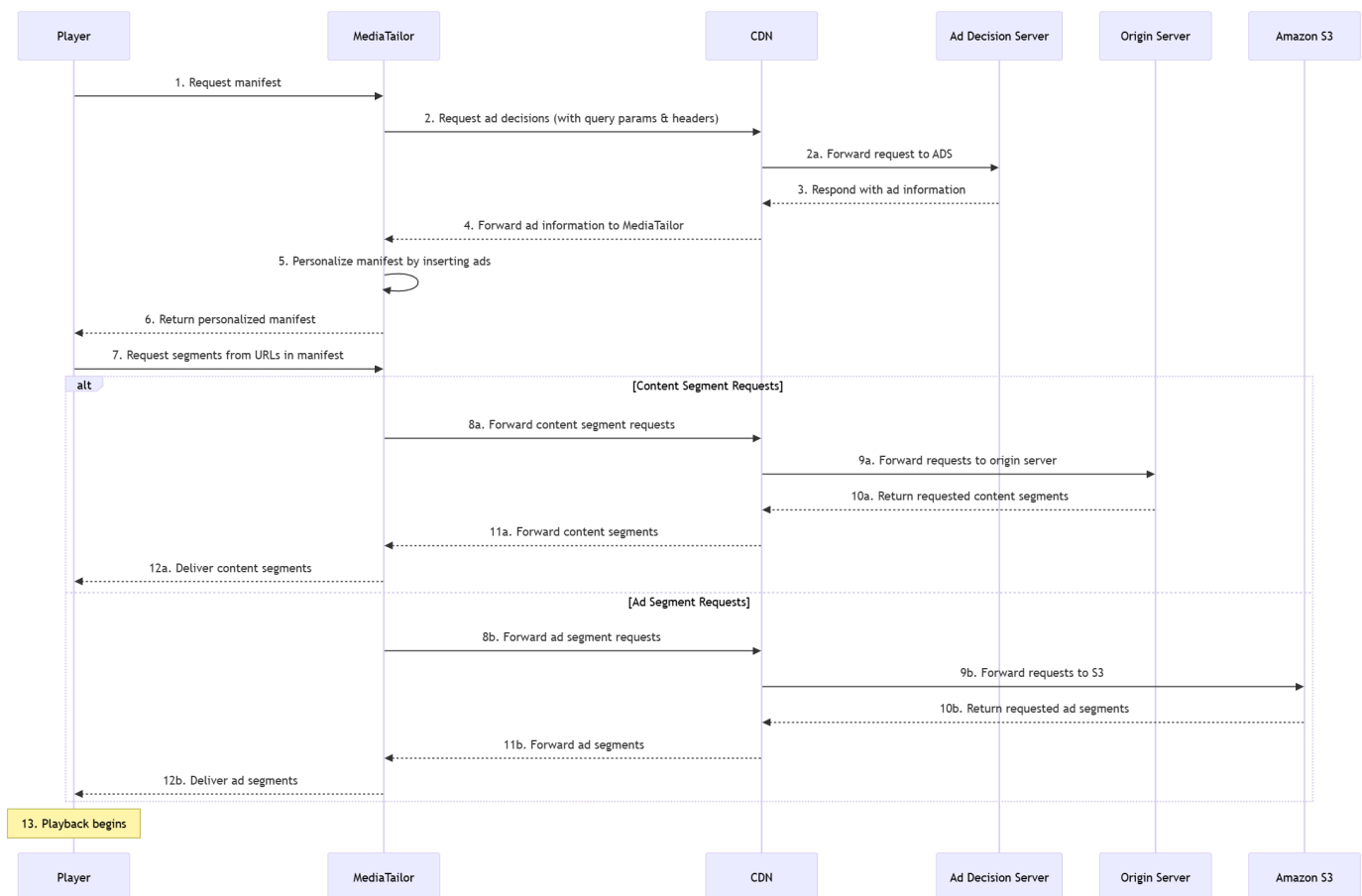
Gerenciamento de cache

Implemente estratégias simplificadas de invalidação de cache, especialmente para conteúdo ativo em que os manifestos são atualizados com frequência.

Nessa arquitetura abaixo do ideal:

1. Os espectadores solicitam playlists multivariantes, playlists de mídia ou diretamente de MPDs AWS Elemental MediaTailor

2. MediaTailor solicita manifestos de conteúdo (playlists multivariantes, playlists de mídia ou) por meio da CDN. MPDs
3. A CDN encaminha as solicitações para o servidor de origem.
4. O servidor de origem retorna playlists multivariantes, playlists de mídia ou para a CDN. MPDs
5. O CDN encaminha playlists multivariantes, playlists de mídia ou para. MPDs MediaTailor
6. MediaTailor solicita anúncios do servidor de decisão de anúncios (ADS).
7. MediaTailor personaliza manifestos inserindo anúncios em playlists multivariantes, playlists de mídia ou os entrega diretamente aos espectadores. MPDs
8. Essa arquitetura introduz latência adicional, possíveis problemas de cache e complica a solução de problemas.



Fluxo de solicitações e respostas

Ao implementar a inserção dinâmica de anúncios com uma CDN, configure seu sistema para suportar esse fluxo de solicitação e resposta:

1. Configure seu player para solicitar playlists multivariantes (HLS) ou MPDs (DASH) do seu CDN como origem do manifesto. MediaTailor
2. Configure sua CDN para encaminhar todas as solicitações multivariadas de playlist, playlist de mídia e MPD MediaTailor, incluindo todos os parâmetros e cabeçalhos de consulta.
3. Certifique-se de MediaTailor poder se comunicar com seu servidor de decisão de anúncios (ADS), transmitindo parâmetros e cabeçalhos de consulta.
4. Configure seu ADS para usar parâmetros de consulta para determinar quais anúncios inserir.
5. Configure o prefixo CDN na configuração de MediaTailor reprodução para MediaTailor poder substituir os nomes de domínio CDN pelos prefixos de URL do conteúdo e do segmento de anúncios.
6. Configure sua CDN para encaminhar playlists multivariantes personalizadas, playlists de mídia e MPDs de para o player solicitante. MediaTailor
7. Configure sua CDN para traduzir segmentos URLs, encaminhando solicitações de segmentos de conteúdo para o servidor de origem e solicitações de anúncios para o bucket do Amazon S3, MediaTailor onde armazena anúncios transcodificados.

Terminologia CDN para inserção de anúncios

A compreensão desses termos-chave ajudará você a implementar e solucionar problemas de integração com a CDN de inserção de anúncios:

CDN de origem e CDN de borda

CDN de origem: uma CDN posicionada entre MediaTailor e a origem do seu conteúdo. Ele armazena segmentos de conteúdo em cache para reduzir a carga em seus servidores de origem. Em uma arquitetura multi-CDN, essa é a primeira camada CDN que interage diretamente com a origem.

Edge CDN: Uma CDN posicionada entre visualizadores e MediaTailor. Ele oferece manifestos e conteúdo personalizados aos espectadores. Em uma arquitetura multi-CDN, essa é a camada CDN mais externa que interage diretamente com os visualizadores.

Termos de configuração do CDN

Comportamento do cache: regras que determinam como uma CDN lida com diferentes tipos de solicitações. Essas regras incluem:

- Configurações de duração do armazenamento em cache
- Configurações de roteamento de origem
- Parâmetros de tratamento de solicitações

TTL (Time To Live): a duração pela qual o conteúdo permanece válido em um cache CDN antes de precisar ser atualizado a partir da origem.

Chave de cache: o identificador exclusivo usado por uma CDN para armazenar e recuperar conteúdo em cache. Normalmente inclui:

- Caminho do URL
- Parâmetros de consulta
- Cabeçalhos selecionados

Escudo de origem: uma camada intermediária de cache entre os pontos de borda da CDN e seu servidor de origem. Isso reduz o número de solicitações até sua origem.

Colapso da solicitação: um recurso de CDN que combina várias solicitações simultâneas para o mesmo conteúdo em uma única solicitação de origem.

MediaTailor-termos específicos da CDN

Prefixo do segmento de conteúdo CDN: O nome de domínio CDN AWS Elemental MediaTailor usado na geração de segmentos de conteúdo em URLs manifestos.

Prefixo do segmento de anúncio CDN: o nome de domínio CDN MediaTailor usado na geração de segmentos de anúncios em URLs manifestos.

Para obter mais informações sobre a configuração de CDN com MediaTailor, consulte [Configurar a integração com a CDN](#).

Note

Esses termos são consistentes com aqueles usados na documentação de montagem do canal. Para a terminologia de montagem de canais, consulte [Terminologia CDN para montagem de canais](#).

Configure o MediaTailor SSAI básico com uma CDN para otimizar a entrega de anúncios

Esta seção fornece step-by-step instruções para configurar a inserção AWS Elemental MediaTailor dinâmica de anúncios com uma rede de distribuição de conteúdo (CDN) para otimizar seu fluxo de trabalho de monetização de vídeos.

Para configuração avançada do servidor de anúncios usando variáveis dinâmicas, consulte [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#). Para obter informações sobre como transmitir parâmetros CDNs para a segmentação de anúncios, consulte [MediaTailor parâmetros de consulta do manifesto](#).

Para obter informações conceituais sobre o SSAI com CDNs, consulte. [Entenda a arquitetura de inserção de anúncios para CDN e integrações MediaTailor](#)

Pré-requisitos

Antes de configurar a inserção de anúncios com uma CDN, verifique se você tem:

- Uma AWS Elemental MediaTailor configuração ativa
- Um servidor de origem de conteúdo que fornece conteúdo HLS ou DASH com marcadores de anúncios apropriados

Para obter informações sobre marcadores de anúncios, consulte [Compreender o comportamento de inserção de anúncios](#).

- Um servidor de decisão de anúncios (ADS) que suporta VAST ou VMAP para segmentação de anúncios
- Uma conta CDN (como Amazon CloudFront ou outro provedor de CDN)
- Conhecimento básico de manipulação de manifestos e conceitos de inserção dinâmica de anúncios

Etapa 1: configurar o cache da CDN para uma entrega ideal de anúncios

A configuração adequada do cache de CDN é fundamental para o desempenho ideal do fluxo de trabalho de monetização de vídeo. Os requisitos de cache diferem entre inserção de anúncios no lado do servidor (SSAI) e inserção de anúncios guiada pelo servidor (SGAI). Use essas configurações recomendadas para garantir a entrega eficiente de conteúdo e publicidade personalizada:

Configurações de cache SSAI CDN

Para fluxos de trabalho de inserção de anúncios do lado do servidor, a configuração adequada do armazenamento em cache é essencial para um desempenho ideal. O SSAI exige valores específicos de TTL e configurações de chave de cache para garantir que manifestos personalizados não sejam armazenados em cache enquanto os segmentos são armazenados em cache de forma eficiente.

Para obter configurações detalhadas de cache SSAI, incluindo valores TTL, padrões de caminho e configurações de chave de cache, consulte o guia de otimização da [Cache de inserção de anúncios do lado do servidor \(SSAI\) CDN](#).

Princípios-chave de armazenamento em cache para SSAI:

- Manifestos: defina TTL para 0 segundos para evitar o armazenamento em cache de conteúdo personalizado
- Segmentos: armazene em cache agressivamente (mais de 24 horas) para reduzir a carga de origem
- Chaves de cache: inclua todos os parâmetros de consulta para manifestos, caminho de URL somente para segmentos

Configurações de cache do SGAI CDN

Para fluxos de trabalho de inserção de anúncios guiados pelo servidor, os requisitos de armazenamento em cache diferem dos da SSAI porque os manifestos do SGAI podem ser armazenados em cache por curtos períodos e, ao mesmo tempo, fornecer experiências publicitárias personalizadas.

Para configurações abrangentes de cache do SGAI, incluindo valores de VOD e TTL ativos, consulte as tabelas de armazenamento em cache do guia de otimização. O SGAI permite uma melhor eficiência de cache do que o SSAI, mantendo os recursos de personalização de anúncios.

Principais diferenças de cache do SGAI:

- Manifestos: podem ser armazenados em cache por curtos períodos (5 a 30 minutos para VOD, 2 a 10 segundos para transmissão ao vivo)
- Segmentos: armazene em cache de forma agressiva, como o SSAI (mais de 24 horas para a maioria dos conteúdos)
- Benefício de desempenho: melhores taxas de acerto de cache do que SSAI devido aos manifestos que podem ser armazenados em cache

Para a Amazon CloudFront, você pode implementar essas configurações usando comportamentos de cache com diferentes valores de TTL e políticas de chave de cache. Por outro lado CDNs, consulte a documentação específica para implementar regras de armazenamento em cache semelhantes.

Etapa 2: Implementar abordagens híbridas (se necessário)

Se sua arquitetura exigir uma abordagem híbrida com uma CDN separada ou uma camada de cache entre a origem do conteúdo e MediaTailor

1. Implemente uma separação clara de interesses em sua configuração de CDN.
2. Defina configurações específicas de CDN para evitar os problemas técnicos descritos na seção anterior.
3. Teste minuciosamente sua configuração para verificar se a personalização do manifesto funciona corretamente.
4. Monitore as métricas de desempenho para garantir a entrega ideal de playlists multivariantes, playlists de mídia e segmentos. MPDs

Ao implementar uma abordagem híbrida, considere estas configurações específicas:

- Para a CDN entre a origem do conteúdo e MediaTailor:
 - Configure a passagem de compactação para arquivos de manifesto para preservar o estado de compactação original de sua origem
 - Inclua todos os parâmetros de consulta na chave de cache
 - Defina valores curtos de TTL para manifestos de conteúdo ao vivo
- Para o CDN entre MediaTailor e visualizadores:
 - Configure tempos de cache mais longos para segmentos de anúncios
 - Conjunto apropriado TTLs para manifestos personalizados
 - Implemente o roteamento de origem adequado para conteúdo versus segmentos de anúncios

Etapa 3: Concluir sua configuração de CDN

Depois de escolher sua arquitetura e entender o fluxo de solicitações, conclua sua configuração seguindo as etapas detalhadas de configuração em [Configurar a integração com a CDN](#).

Para provedores de CDN específicos, consulte estes recursos adicionais:

- Amazon CloudFront: consulte [Integração do CloudFront](#) as etapas CloudFront de configuração específicas
- Outros CDNs: aplique os princípios gerais descritos neste guia, adaptando-os às opções de configuração específicas da sua CDN

Etapa 4: verificar sua configuração

Depois de concluir a configuração da CDN, verifique se o fluxo de trabalho dinâmico de inserção de anúncios está funcionando corretamente:

1. Teste a reprodução por meio de seu CDN com um reprodutor de amostra
2. Verifique se a publicidade personalizada está inserida corretamente nos pontos de interrupção do anúncio designados
3. Verifique os registros do CDN para confirmar o roteamento adequado da solicitação
4. Monitore as taxas de acerto do cache para garantir o desempenho ideal tanto para o conteúdo quanto para os segmentos de anúncios
5. Confirme se os parâmetros de segmentação de anúncios estão sendo transmitidos corretamente pelo fluxo de trabalho

Para obter procedimentos abrangentes de teste e validação, consulte [Teste e validação para CDN e integrações MediaTailor](#). Para obter informações detalhadas sobre como monitorar sua implementação de SSAI, consulte [Monitore as operações de CDN e MediaTailor integrações](#). Para otimizar o desempenho, consulte [Otimize o desempenho para CDN e integrações MediaTailor](#).

Integre o MediaTailor SSAI com a montagem de canais para canais lineares monetizados

Este tópico explica como combinar a inserção de anúncios no AWS Elemental MediaTailor servidor com a montagem de canais e a integração da rede de distribuição de conteúdo (CDN). Essa integração permite que você:

- Crie canais lineares monetizados com publicidade personalizada
- Ofereça anúncios segmentados para diferentes espectadores que assistem ao mesmo conteúdo
- Mantenha experiências de visualização com qualidade de transmissão

Benefícios da combinação do SSAI com a montagem do canal

A integração do SSAI com a montagem de canais oferece vários benefícios importantes:

Monetização de canais lineares

Insira anúncios personalizados em seus canais lineares para gerar receita com sua biblioteca de conteúdo. Você pode monetizar conteúdo ao vivo e VOD em um único fluxo linear.

Publicidade personalizada

Ofereça anúncios diferentes para espectadores diferentes que assistem ao mesmo conteúdo do canal. Essa abordagem direcionada aumenta a relevância do anúncio e a receita potencial em comparação com a publicidade televisiva tradicional.

Gerenciamento simplificado de intervalos publicitários

Defina pontos de interrupção de anúncios em seus programas de montagem de canais sem precisar condicionar o conteúdo com marcadores SCTE-35. Isso facilita a inserção de anúncios em pontos de interrupção naturais em seu conteúdo.

Experiência com qualidade de transmissão

MediaTailor mantém uma experiência de visualização de alta qualidade com transições perfeitas entre conteúdo e anúncios. A inserção de anúncios no lado do servidor elimina muitos problemas comuns, como:

- Armazenamento em buffer durante as transições de anúncios
- Bloqueadores de anúncios que impedem a monetização
- Qualidade de reprodução inconsistente

Entrega escalável

Quando combinada com uma CDN, essa integração pode ser escalada para milhões de visualizadores simultâneos sem degradação no desempenho ou nos recursos de personalização.

Visão geral da arquitetura

A arquitetura para combinar SSAI com montagem de canais normalmente envolve esses componentes:

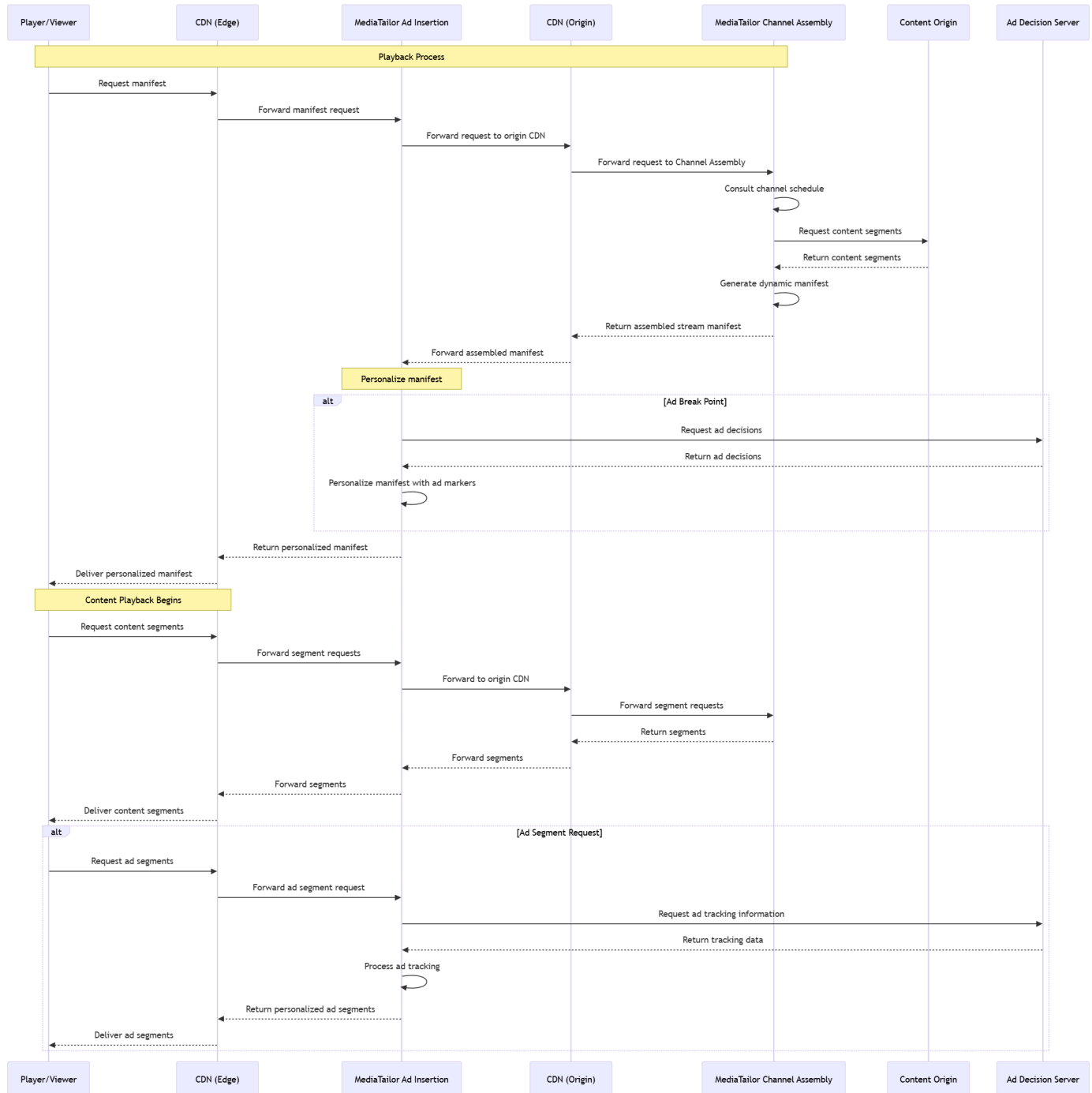
- Montagem de canais: cria canais lineares a partir de VOD e conteúdo ao vivo e insere conteúdo de ardósia que cria marcadores de anúncios no manifesto gerado

- Inserção de anúncios: reconhece pontos de quebra de anúncios e inserções que URLs apontam para segmentos de anúncios personalizados no manifesto
- Servidor de decisão de anúncios (ADS): determina quais anúncios inserir para cada espectador
- Rede de entrega de conteúdo (CDN): entrega o conteúdo reunido e os segmentos de anúncios aos espectadores
- Servidor de origem: armazena os segmentos de VOD e conteúdo ao vivo

Nesta arquitetura:

1. A montagem do canal cria um canal linear a partir do seu VOD e do seu conteúdo ao vivo e insere conteúdo em formato de ardósia que cria os marcadores de anúncios no manifesto gerado
2. Quando um espectador solicita o canal, a inserção do anúncio reconhece as quebras de anúncio que foram inseridas no canal linear
3. A inserção de anúncios liga para o ADS para receber a lista de anúncios, os transcodifica e insere as inserções que URLs apontam para os segmentos de anúncios transcodificados no manifesto personalizado
4. O CDN fornece o fluxo personalizado para o espectador

O diagrama a seguir ilustra esse fluxo de trabalho:



Configurando a integração

Para configurar o SSAI com montagem de canais:

1. Configure sua CDN de borda para aceitar solicitações de manifesto dos espectadores e encaminhá-las para a inserção de AWS Elemental MediaTailor anúncios.
2. Configure a inserção de MediaTailor anúncios para encaminhar solicitações para sua CDN de origem.
3. Configure sua CDN de origem para encaminhar solicitações para a montagem MediaTailor do canal.
4. Configure a montagem MediaTailor do canal para gerar manifestos dinâmicos com base na programação atual.
5. Configure sua CDN de origem para encaminhar os manifestos reunidos para a inserção de MediaTailor anúncios.
6. Configure a inserção de MediaTailor anúncios para solicitar decisões de anúncios do seu servidor de decisão de anúncios nos pontos de interrupção do anúncio.
7. Configure a inserção de MediaTailor anúncios para personalizar manifestos substituindo os marcadores de anúncios (da montagem do canal) por segmentos de URLs anúncios direcionados (do ADS).
8. Configure sua CDN de ponta para oferecer manifestos personalizados aos espectadores.
9. Configure sua arquitetura de CDN para lidar com solicitações de conteúdo e segmento de anúncios de forma eficiente.

Definindo intervalos publicitários na montagem do canal

Ao criar programas na montagem de canais, você pode definir intervalos publicitários de várias maneiras:

Transições do programa

Insira anúncios entre os programas na programação do seu canal. Essa é a abordagem mais simples e garante que os anúncios não interrompam o conteúdo do programa.

Marcadores SCTE-35

Se seu conteúdo VOD contiver marcadores SCTE-35, a montagem do canal poderá preservar esses marcadores e a inserção de anúncios poderá usá-los como pontos de interrupção do anúncio.

Inserção com base no tempo

Defina intervalos publicitários em momentos específicos nos programas. Isso permite que você insira anúncios em pontos de interrupção naturais em seu conteúdo.

Para obter informações detalhadas sobre a criação de programas com intervalos publicitários, consulte Como [trabalhar com programas](#).

Considerações sobre o armazenamento em cache da CDN

Para um desempenho ideal ao combinar montagem de canais e SSAI com uma CDN:

- Configure comportamentos de cache que diferenciem entre montagem de canais e solicitações SSAI
- Defina valores TTL apropriados para manifestos e segmentos, conforme recomendado em [Etapa 1: configurar o cache da CDN para uma entrega ideal de anúncios](#)
- Garanta o roteamento adequado entre a montagem do canal, a inserção do anúncio e as origens da sua CDN
- Monitore as métricas de desempenho dos componentes de montagem de canais e inserção de anúncios

Configurações de cache recomendadas para implementação combinada

Tipo de conteúdo	TTL	Elementos principais do cache
Manifestos de montagem de canais	0 segundos	Caminho do URL + parâmetros de consulta
Manifestos personalizados da SSAI	0 segundos	Caminho do URL + todos os parâmetros de consulta
Segmentos de conteúdo	Mais de 24 horas	Somente caminho de URL
Segmentos de anúncios	Mais de 24 horas	Somente caminho de URL

Monitorando a solução integrada

Para garantir que sua solução integrada tenha um desempenho ideal, monitore essas métricas principais:

Métricas de montagem de canais

Monitore o tempo de geração do manifesto, as transições do programa e quaisquer erros no processo de montagem do canal.

Métricas de inserção de anúncios

Monitore a taxa de preenchimento de anúncios, os tempos de resposta do servidor de decisão de anúncios e os erros de inserção de anúncios.

Métricas de CDN

Monitore as taxas de acerto do cache, o volume da solicitação de origem e a latência de resposta para conteúdo e segmentos de anúncios.

Métricas de experiência do espectador

Monitore eventos de rebuffer, horários de inicialização e engajamento dos espectadores, especialmente durante as transições de anúncios.

Para obter informações detalhadas sobre monitoramento, consulte [Monitore as operações de CDN e MediaTailor integrações](#) [Monitore as operações de CDN de montagem de MediaTailor canais](#) e.

Solução de problemas comuns do

Ao solucionar problemas com a solução integrada, considere estes problemas comuns:

Problemas de sincronização do Ad Break

Se os anúncios não estiverem aparecendo nos pontos de interrupção esperados, verifique se as definições de quebra de anúncio nos programas de montagem de canais estão configuradas corretamente e se a inserção do anúncio está identificando adequadamente esses pontos de interrupção.

Erros de entrega manifestos

Se os espectadores estiverem enfrentando problemas de reprodução, verifique se a CDN está encaminhando corretamente as solicitações de manifesto entre a montagem do canal e a

inserção de anúncios e se as configurações de cache são apropriadas para a natureza dinâmica desses manifestos.

Problemas de roteamento de segmentos

Se o conteúdo ou os segmentos de anúncios não estiverem sendo carregados, verifique se a CDN está encaminhando corretamente as solicitações do segmento para as origens apropriadas e se o segmento URLs nos manifestos está formatado corretamente.

Degradação do desempenho

Se os espectadores estiverem enfrentando buffering ou alta latência, verifique as taxas de acerto do cache da CDN e os volumes de solicitações de origem para identificar possíveis gargalos no pipeline de entrega.

Para obter mais orientações sobre a solução de problemas, consulte [Solucione problemas de MediaTailor SSAI com a entrega ininterrupta CDNs de anúncios](#).

Práticas recomendadas

Siga estas melhores práticas para uma integração bem-sucedida do SSAI com a montagem de canais:

- **Teste minuciosamente:** teste a solução integrada com vários tipos de conteúdo, cenários de anúncios e condições de visualização antes de implantá-la na produção.
- **Monitore continuamente:** configure monitoramento e alertas abrangentes para identificar e resolver rapidamente quaisquer problemas que surjam.
- **Otimize o armazenamento em cache:** revise e ajuste regularmente as configurações de cache da CDN com base nos padrões reais de uso e nas métricas de desempenho.
- **Planeje a escala:** projete sua arquitetura para lidar com picos de tráfego, especialmente para canais ou eventos populares.
- **Considere a redundância:** implemente redundância em componentes críticos para garantir a alta disponibilidade de seus canais lineares.
- **Otimize as transições de anúncios:** garanta transições suaves entre conteúdo e anúncios usando perfis de codificação e durações de segmentos consistentes.

Informações relacionadas

Para obter mais informações sobre a integração do SSAI com a montagem de canais, consulte:

Documentação de montagem do canal

[Usando AWS Elemental MediaTailor para criar fluxos montados lineares](#)- Aprenda sobre os conceitos de montagem de canais

[Montagem de canais com CDN](#)- Configure a montagem do canal com uma CDN

Documentação da SSAI

[Inserção de anúncios com CDN](#)- Configurar a inserção de anúncios com um CDN

[Entenda a arquitetura de inserção de anúncios para CDN e integrações MediaTailor](#)- Entenda a arquitetura CDN de inserção de anúncios

Configuração da CDN

[Configurar a integração com a CDN](#)- Orientação geral de configuração da CDN

[Integração do CloudFront](#)- configuração CloudFront específica

Otimize o desempenho para CDN e integrações MediaTailor

Maximize o desempenho de sua implementação de inserção de AWS Elemental MediaTailor anúncios otimizando sua configuração de rede de distribuição de conteúdo (CDN). Essas configurações garantem uma entrega eficiente de conteúdo e uma experiência ideal para o espectador.

Para obter orientações detalhadas de otimização de cache e roteamento específicas para implementações de SSAI, consulte. [Otimização do desempenho da CDN](#) O guia de otimização consolidado fornece configurações abrangentes de cache, configuração de roteamento de solicitações e benchmarks de desempenho que se aplicam a todas as integrações de CDN. MediaTailor

Desafios comuns de desempenho

As implementações da SSAI CDNs podem enfrentar vários desafios de desempenho:

Sobrecarga de manipulação do manifesto

MediaTailor executa a manipulação do manifesto em tempo real, que pode introduzir latência se não for otimizada adequadamente. O seguinte pode introduzir latência:

- Tempo de processamento para solicitações do Ad Decision Server (ADS)
- Tempo necessário para modificar manifestos com referências de segmentos de anúncios
- Processamento adicional para personalização

Problemas de eficiência de cache

Manifestos personalizados podem reduzir a eficiência do cache da CDN porque:

- Cada espectador pode receber um manifesto exclusivo
- Os parâmetros da sessão podem fragmentar o cache
- O conteúdo dinâmico requer uma configuração cuidadosa do cache

Picos de carga do Origin

O armazenamento em cache incorreto pode levar a picos de carga de origem durante:

- Eventos de alto tráfego
- Atualizações de cache
- Alterações na configuração do CDN

Problemas de reprodução relacionados a anúncios

A inserção de anúncios pode causar interrupções na reprodução, como:

- Armazenamento em buffer durante as transições de anúncios
- Diferenças de qualidade entre conteúdo e anúncios
- Falhas na reprodução quando os anúncios não podem ser recuperados

Para obter uma orientação abrangente sobre otimização de desempenho, incluindo estratégias de armazenamento em cache, roteamento de solicitações, benchmarks de desempenho e técnicas avançadas de otimização, consulte [Otimização do desempenho da CDN](#). O guia de otimização consolidado fornece configurações e benchmarks detalhados que se aplicam a todas as integrações de MediaTailor CDN, incluindo implementações de SSAI.

Resumo das melhores práticas

Siga essas melhores práticas para garantir o desempenho ideal do SSAI com CDNs:

Práticas recomendadas de arquitetura

- Escolha o padrão arquitetônico certo para sua escala e seus requisitos

- Implemente serviços em estreita proximidade para minimizar a latência
- Implemente redundância para componentes críticos

Práticas recomendadas de armazenamento em cache

- Use diferentes estratégias de armazenamento em cache para diferentes tipos de conteúdo
- Otimize as chaves de cache para equilibrar personalização e eficiência
- Defina o adequado TTLs com base no tipo de conteúdo e na frequência de atualização. Para obter recomendações detalhadas de TTL, consulte [Otimização de cache para CDN e integrações MediaTailor](#).

Práticas recomendadas de entrega de anúncios

- Otimize as interações do ADS com tempos limite e recuos
- Prepare anúncios de acordo com as especificações do conteúdo
- Implemente uma entrega eficiente de segmentos de anúncios

Monitorando as melhores práticas

- Monitore todos os componentes da sua implementação de SSAI
- Configure alertas para degradação do desempenho
- Revise e otimize regularmente sua configuração

Orientação completa de otimização

Para obter uma orientação abrangente de otimização de CDN, incluindo estratégias detalhadas de armazenamento em cache, configuração de roteamento de solicitações, benchmarks de desempenho e técnicas avançadas de otimização, consulte. [Otimização do desempenho da CDN](#) O guia de otimização consolidado fornece configurações e benchmarks completos que se aplicam a todas as integrações de MediaTailor CDN, incluindo implementações de SSAI.

Monitore as operações de CDN e MediaTailor integrações

AWS Elemental MediaTailor fornece recursos analíticos robustos que, quando combinados com métricas de rede de entrega de conteúdo (CDN), oferecem insights abrangentes sobre sua implementação de SSAI. Este tópico aborda:

Para obter uma orientação abrangente de monitoramento de CDN, incluindo métricas essenciais, configuração de ferramentas de monitoramento, configuração de alertas e estratégias de solução de problemas que se aplicam a todas as MediaTailor implementações, consulte. [Monitoramento de](#)

[CDN](#) Este tópico se concentra nos requisitos de monitoramento específicos do SSAI e na análise de inserção de anúncios.

- Estratégias de monitoramento para integração de SSAI e CDN
- Ferramentas de análise e métodos de coleta de dados
- Técnicas de otimização orientadas por dados

Principais métricas para SSAI com CDNs

Para monitorar com eficácia sua implementação de SSAI com CDNs, acompanhe estas métricas essenciais:

Métricas de inserção de anúncios

Taxa de preenchimento do anúncio: a porcentagem de oportunidades de anúncios que foram preenchidas com sucesso com anúncios.

Taxa de erro do anúncio: a porcentagem de solicitações de anúncios que resultaram em erros.

Tempo de resposta do anúncio: quanto tempo leva para o servidor de decisão de anúncios responder às solicitações de anúncios.

Precisão da duração do anúncio: até que ponto a duração real dos anúncios inseridos corresponde à duração esperada.

Métricas de experiência do espectador

Taxa de rebuffer: a porcentagem do tempo de visualização gasto no buffer.

Tempo de inicialização: quanto tempo leva para a reprodução do vídeo começar.

Suavidade da transição de anúncios: a facilidade com que o player faz a transição entre conteúdo e anúncios.

Duração da sessão: quanto tempo os espectadores assistem antes de abandonarem a transmissão.

Ferramentas de análise e integração

Combine essas ferramentas para criar uma solução de análise abrangente para sua implementação de SSAI:

AWS Elemental MediaTailor métricas do lado do servidor

MediaTailor fornece métricas integradas por meio da Amazon CloudWatch que rastreiam solicitações, respostas e erros de anúncios. Essas métricas podem ser visualizadas no CloudWatch console ou integradas em painéis personalizados.

MediaTailor As principais métricas incluem:

- `AdDecisionServer.Ads`: o número de anúncios retornados pelo servidor de decisão de anúncios.
- `AdDecisionServer.Duration`: a duração total dos anúncios retornados pelo servidor de decisão de anúncios.
- `AdDecisionServer.Errors`: o número de erros retornados pelo servidor de decisão do anúncio.
- `AdDecisionServer.Latency`: o tempo de resposta do servidor de decisão do anúncio.

Para obter uma lista completa de MediaTailor métricas, consulte [Monitoramento MediaTailor com a Amazon CloudWatch](#).

Análise de CDN

Os provedores de CDN oferecem análises detalhadas sobre o desempenho da entrega de conteúdo. Para a Amazon CloudFront, use CloudWatch métricas e registros de CloudFront acesso da Amazon para analisar padrões de entrega.

Métricas CDN importantes para monitorar:

- Contagem de solicitações por tipo de conteúdo (manifestos versus segmentos)
- Taxa de acerto do cache para diferentes tipos de conteúdo
- Distribuição geográfica dos espectadores
- Taxas de erro por código de erro

Rastreamento do lado do cliente

Implemente o rastreamento do lado do cliente para coletar métricas de experiência do espectador que não estão disponíveis no lado do servidor:

- Eventos do jogador (reprodução, pausa, busca, buffer)
- Taxas de conclusão de visualizações de anúncios
- Métricas de qualidade de serviço (alterações de resolução, taxa de bits)
- Padrões de engajamento dos espectadores

Considere usar o rastreamento MediaTailor do lado do cliente para coletar e relatar essas métricas.

Painéis integrados

Crie painéis abrangentes que combinam métricas de várias fontes:

- Use CloudWatch painéis para combinar MediaTailor métricas CloudFront
- Considere plataformas de análise de terceiros para uma visualização mais avançada
- Configure a correlação entre serviços para identificar relacionamentos entre métricas

Implementando uma estratégia de monitoramento

Siga estas etapas para implementar uma estratégia abrangente de monitoramento para sua implementação de SSAI com CDN:

1. Configurar o monitoramento básico

- Ative CloudWatch métricas para MediaTailor
- Configurar o registro de CDN e a coleta de métricas
- Implemente o rastreamento do lado do cliente em seu player de vídeo

2. Criar painéis personalizados

- Crie um CloudWatch painel que combine as principais métricas
- Inclua visualizações para taxa de preenchimento de anúncios, desempenho de CDN e experiência do espectador
- Adicione anotações para eventos importantes (alterações de configuração, transmissões importantes)

3. Configurar alertas

- Configure CloudWatch alarmes para métricas críticas
- Crie alarmes compostos que são acionados em várias condições relacionadas
- Configurar canais de notificação (e-mail, SMS, Amazon SNS)

4. Implemente respostas automatizadas

- Use CloudWatch Eventos para acionar respostas automatizadas para problemas comuns
- Crie runbooks para intervenção manual quando necessário
- Documente os procedimentos de solução de problemas para diferentes cenários de alerta

Exemplo Criação de um painel de monitoramento SSAI abrangente

Este exemplo mostra como criar um CloudWatch painel que combina CloudFront métricas MediaTailor e:

```
{
  "widgets": [
    {
      "type": "metric",
      "properties": {
        "metrics": [
          [ "AWS/MediaTailor", "AdDecisionServer.Ads", "Configuration", "your-config-name" ],
          [ ".", "AdDecisionServer.Errors", ".", "." ]
        ],
        "period": 300,
        "stat": "Sum",
        "region": "us-west-2",
        "title": "Ad Decision Server Performance"
      }
    },
    {
      "type": "metric",
      "properties": {
        "metrics": [
          [ "AWS/CloudFront", "Requests", "DistributionId", "your-distribution-id" ],
          [ ".", "4xxErrorRate", ".", "." ],
          [ ".", "5xxErrorRate", ".", "." ]
        ],
        "period": 300,
        "stat": "Average",
        "region": "us-east-1",
        "title": "CDN Performance"
      }
    }
  ]
}
```

Otimização baseada em dados

Use os dados de análise que você coleta para otimizar sua implementação de SSAI:

Otimização do cache CDN

Analise as taxas de acerto do cache para identificar oportunidades de melhoria:

- Ajuste as configurações de TTL com base no tipo de conteúdo e na frequência de atualização. Para obter recomendações detalhadas de TTL, consulte [Etapa 1: configurar o cache da CDN para uma entrega ideal de anúncios](#).
- Otimize as configurações da chave de cache para melhorar a eficiência do cache
- Considere implementar o escudo de origem para armazenamento em cache em várias camadas

Otimização da entrega de anúncios

Use métricas de desempenho de anúncios para melhorar a entrega de anúncios:

- Identifique e resolva erros comuns de inserção de anúncios
- Otimize os tempos de resposta do servidor de decisões publicitárias
- Ajuste os parâmetros de segmentação de anúncios com base na análise da taxa de preenchimento

Otimização da experiência do espectador

Melhore a experiência do espectador com base nas métricas do lado do cliente:

- Analise os padrões de desistência durante os intervalos publicitários
- Otimize os pontos de transição do anúncio para uma reprodução mais suave
- Ajuste a frequência e a duração do anúncio com base nos dados de engajamento do espectador

Otimização de custos

Equilibre as considerações de desempenho e custo:

- Analise os padrões de uso da largura de banda para otimizar os custos da CDN
- Considere ajustes de classe de preço para CloudFront distribuições
- Avalie o custo-benefício de diferentes estratégias de armazenamento em cache

Práticas recomendadas

Siga estas melhores práticas para monitoramento e análise eficazes da SSAI:

- Estabeleça linhas de base: colete métricas durante a operação normal para estabelecer linhas de base de desempenho que possam ser usadas para comparação durante a solução de problemas.

- Implemente monitoramento em vários níveis: monitore em diferentes níveis de sua arquitetura (origem, CDN, player) para obter uma visão completa do desempenho.
- Correlacione métricas entre serviços: procure relacionamentos entre métricas de diferentes serviços para identificar as causas-raiz dos problemas.
- Use a detecção de anomalias: implemente a detecção de CloudWatch anomalias para identificar automaticamente padrões incomuns em suas métricas.
- Revise e refine regularmente: agende revisões regulares de sua estratégia de monitoramento e ajuste com base nas mudanças nos requisitos e nos novos insights.
- Documente descobertas e ações: mantenha um registro dos esforços de otimização e seus resultados para construir conhecimento institucional.

Informações relacionadas

Para obter mais informações sobre monitoramento e análise para SSAI com CDNs, consulte:

- [Otimize o desempenho para CDN e integrações MediaTailor](#) para técnicas de otimização de desempenho
- [Solucione problemas de MediaTailor SSAI com a entrega ininterrupta CDNs de anúncios](#) para solucionar problemas comuns
- [Monitoramento MediaTailor com CloudWatch a Amazon](#) para obter informações detalhadas sobre MediaTailor métricas
- [Visualização CloudFront e métricas de função de borda](#) para obter informações sobre CloudFront métricas

Solucione problemas de MediaTailor SSAI com a entrega ininterrupta CDNs de anúncios

Esta seção fornece soluções para problemas comuns ao usar a inserção AWS Elemental MediaTailor dinâmica de anúncios com uma rede de distribuição de conteúdo (CDN). Essas soluções ajudarão você a solucionar problemas com a monetização de seu vídeo por meio de publicidade personalizada.

Para obter uma orientação abrangente sobre solução de problemas de CDN, incluindo problemas de desempenho de cache, resolução de erros HTTP, procedimentos de teste e técnicas de diagnóstico que se aplicam a todas as MediaTailor implementações, consulte. [Solucionar problemas de](#)

[Integração com a CDN](#) Esta seção se concentra nos requisitos de solução de problemas específicos do SSAI e nos problemas de inserção de anúncios.

Se você encontrar problemas com sua configuração de CDN e SSAI, verifique estes problemas comuns:

A publicidade personalizada não aparece na transmissão

Verifique se seu ADS responde corretamente e se AWS Elemental MediaTailor comunica com ele. Verifique esses possíveis problemas:

- Parâmetros de consulta de segmentação de anúncios não foram encaminhados adequadamente por meio de sua CDN
- Pontos de interrupção do anúncio definidos incorretamente em seu conteúdo
- Problemas de conectividade ou resposta do ADS

Erros de reprodução em pontos de interrupção do anúncio

Certifique-se de que os segmentos de anúncios sejam transcodificados adequadamente para corresponder às taxas de bits e resoluções do seu conteúdo. Verifique esses problemas comuns:

- CDN roteando solicitações incorretamente para segmentos de anúncios
- Erros manifestos de manipulação em pontos de transição
- Perfis de codificação incompatíveis entre conteúdo e anúncios

Manifestos obsoletos

Para conteúdo ao vivo, verifique se as configurações TTL do cache CDN são apropriadas. Para manifestos personalizados, use TTL de 0 segundos. Considere implementar a invalidação de cache para manifestos que mudam rapidamente. Para obter uma orientação abrangente sobre TTL, consulte [Otimização de cache para CDN e integrações MediaTailor](#).

Alta latência

Verifique sua configuração de CDN para obter o roteamento ideal. Certifique-se de que sua CDN tenha pontos de presença próximos aos seus espectadores para obter o melhor desempenho.

Solução de problemas avançada

Para problemas mais complexos, experimente estas técnicas avançadas de solução de problemas:

Comportamento inconsistente de segmentação de anúncios

Verifique se há inconsistências nos parâmetros de consulta entre as solicitações do jogador e as solicitações do ADS. Certifique-se de que todos os parâmetros de segmentação necessários sejam passados corretamente pela CDN.

Inconsistências do cache CDN

Verifique as configurações da chave de cache para garantir a diferenciação adequada do conteúdo. Considere implementar a limpeza do cache para atualizações críticas do manifesto.

Falhas no rastreamento de anúncios

Verifique se os beacons foram URLs encaminhados corretamente e não foram bloqueados pela CDN. Verifique se os jogadores do cliente podem alcançar os endpoints de rastreamento.

Otimização de desempenho

Para otimizar o desempenho do seu fluxo de trabalho dinâmico de inserção de anúncios e monetização de vídeos:

- Ajuste as configurações de TTL com base no tipo de conteúdo e nos padrões do visualizador. Para obter recomendações detalhadas de TTL, consulte [Otimização de cache para CDN e integrações MediaTailor](#).
- Implemente o roteamento geográfico para minimizar a latência para públicos globais
- Considere o uso de vários CDNs para redundância e desempenho ideal
- Monitore as taxas de acerto do cache e ajuste as configurações adequadamente
- Otimize os processos de manipulação de manifestos para reduzir o tempo de processamento nos pontos de interrupção do anúncio
- Pré-transcodifique anúncios para corresponder a perfis de conteúdo comuns para inserção perfeita de publicidade personalizada

Benchmarks de desempenho

Ao otimizar sua integração CDN de inserção de AWS Elemental MediaTailor anúncios, busque estes benchmarks de desempenho:

Metas de taxa de acertos do cache

Segmentos de conteúdo: maior que 95% de taxa de acertos do cache

Segmentos de anúncios: taxa de acerto de cache superior a 90%

Manifestos: Não aplicável (não devem ser armazenados em cache para inserção de anúncios personalizados)

Benchmarks de latência

Latência da solicitação de manifesto: menos de 100 ms (P95)

Entrega do segmento de conteúdo: menos de 50 ms (P95)

Entrega do segmento de anúncios: menos de 75 ms (P95)

End-to-end tempo de inicialização: menos de 2 segundos

Métricas de carga de origem

Solicitações de origem por espectador: menos de 0,1 solicitações por minuto por espectador

Largura de banda de origem por espectador: menos de 5% da largura de banda total do espectador

Metas de taxa de erro

Erros manifestos: menos de 0,1%

Erros de segmento: menos de 0,01%

Rebuffer relatado pelo jogador: menos de 1%

Benchmarks de escalabilidade

Support para 10 vezes o tráfego normal durante eventos de pico sem degradação

Capacidade de lidar com mais de 1000 solicitações por segundo por canal

Use CloudWatch as métricas da Amazon para monitorar esses indicadores de desempenho. Para obter instruções detalhadas de monitoramento, consulte [Monitoramento AWS Elemental MediaTailor com CloudWatch métricas da Amazon](#).

Informações relacionadas

Para obter mais informações sobre a inserção de anúncios com CDNs, consulte:

Documentação de inserção de anúncios

[Introdução à inserção MediaTailor de anúncios](#)- Saiba mais sobre os conceitos de inserção de anúncios

[Configurar](#)- Comece com a inserção de anúncios

Integração CDN

[Configurar a integração com a CDN](#)- Orientação geral de configuração da CDN

[Integração do CloudFront](#)- configuração CloudFront específica

Integração de montagem de canais

[Montagem de canais com CDN](#)- Saiba mais sobre a montagem de canais com CDNs

[Implementar inserção de anúncios](#)- Implementar a inserção de anúncios com montagem de canais

Monitoramento e otimização

[Monitore as operações de CDN e MediaTailor integrações](#)- Monitoramento e análise abrangentes

[Otimize o desempenho para CDN e integrações MediaTailor](#)- Guia de otimização de desempenho

[Monitoramento AWS Elemental MediaTailor com CloudWatch métricas da Amazon](#)- CloudWatch métricas para MediaTailor

Crie canais MediaTailor lineares com montagem de canais e CDN

Esta seção fornece uma orientação abrangente para integrar a montagem de AWS Elemental MediaTailor canais com uma rede de distribuição de conteúdo (CDN). Siga estas etapas para configurar, configurar e otimizar sua integração de CDN de montagem de canais.

Você também pode combinar a montagem de canais com a inserção de anúncios no lado do servidor (SSAI) para criar canais lineares monetizados com publicidade personalizada. Essa integração poderosa permite que você forneça anúncios direcionados para diferentes espectadores que assistem ao mesmo conteúdo do canal, aumentando suas oportunidades de receita e mantendo uma experiência de visualização com qualidade de transmissão. Para obter informações sobre o SSAI com CDNs, consulte [Inserção de anúncios com CDN](#).

Neste tópico, usamos o termo manifestos para nos referirmos coletivamente a playlists multivariantes, playlists de mídia e. MPDs

Para obter mais informações sobre montagem de MediaTailor canais, consulte [Usando AWS Elemental MediaTailor para criar fluxos montados lineares](#).

O que você precisará

Antes de configurar a montagem MediaTailor do canal com uma CDN, reúna os seguintes recursos necessários:

AWS conta e permissões

Uma AWS conta com permissões apropriadas para criar e gerenciar MediaTailor recursos

Permissões do IAM para MediaTailor, CloudFront (se estiver usando) e serviços relacionados

Para obter os requisitos de permissão detalhados, consulte [Segurança em AWS Elemental MediaTailor](#).

Serviços necessários

Um MediaTailor canal de montagem de canais em execução (não apenas uma configuração SSAI)

Uma conta de rede de distribuição de conteúdo (CDN) (Amazon CloudFront ou CDN de terceiros)

Armazenamento de origem para seu conteúdo VOD (Amazon S3 MediaPackage ou outro servidor de origem)

Requisitos de conteúdo

Fontes de VOD devidamente codificadas e empacotadas no formato HLS ou DASH. Para obter informações sobre como trabalhar com locais de origem e fontes de VOD, consulte [Trabalhando com locais de origem](#).

Conteúdo com durações de segmentos consistentes (mínimo recomendado: 1 segundo)

Conteúdo da lista de anúncios para intervalos publicitários (se estiver implementando a inserção de anúncios). Para obter informações sobre como configurar o slate, consulte. [MediaTailor ardósia e inserção](#)

Antes de começar

Important

Esse fluxo de trabalho requer um MediaTailor canal de montagem de canais em execução. Ter somente uma configuração MediaTailor SSAI não é suficiente para essa integração. Você deve ter um canal de montagem de canais ativo configurado e operacional antes de prosseguir com a integração do CDN.

Conclua essas tarefas de configuração antes de implementar a montagem do MediaTailor canal com uma CDN:

1. Configure a conectividade de rede entre seu CDN e MediaTailor servidores de origem
2. Configure HTTPS para entrega segura de conteúdo
3. Defina as configurações de DNS para seu domínio CDN

Pré-requisitos de conhecimento

Para implementar essa solução com sucesso, você deve ter:

- Compreensão dos protocolos de streaming (HLS/DASH)
- Conhecimento básico dos princípios de configuração da CDN
- Familiaridade com os conceitos de montagem de MediaTailor canais

Para MediaTailor configuração básica, consulte [Configurar Introdução à montagem MediaTailor de canais](#) e. Para obter informações sobre como trabalhar com locais de origem e fontes de VOD, consulte [Trabalhando com locais de origem](#). Para obter informações sobre como configurar o slate, consulte. [MediaTailor ardósia e inserção](#)

Benefícios da integração com a CDN

A integração da montagem de canais com uma CDN oferece esses principais benefícios.

Experiência aprimorada do espectador

A configuração adequada CDNs reduz o buffer, os tempos de inicialização e os erros de reprodução para canais lineares. Isso resulta em maior engajamento e satisfação do espectador.

Redução de custos

Estratégias eficientes de armazenamento em cache minimizam as solicitações de origem. Isso reduz os custos de transferência de dados e a carga do servidor de origem, particularmente importante para canais lineares de alto volume.

Escalabilidade

As configurações otimizadas de CDN lidam com picos de tráfego durante eventos populares sem degradar o desempenho, garantindo que seus canais lineares permaneçam disponíveis mesmo nos horários de pico de visualização.

Alcance global

Configurado corretamente, CDNs forneça conteúdo com baixa latência para espectadores em todo o mundo, independentemente de sua localização, expandindo seu público potencial.

Transições de programa perfeitas

A configuração otimizada de CDN garante transições suaves entre programas em seu canal linear, criando uma experiência de visualização com qualidade de transmissão.

Tópicos

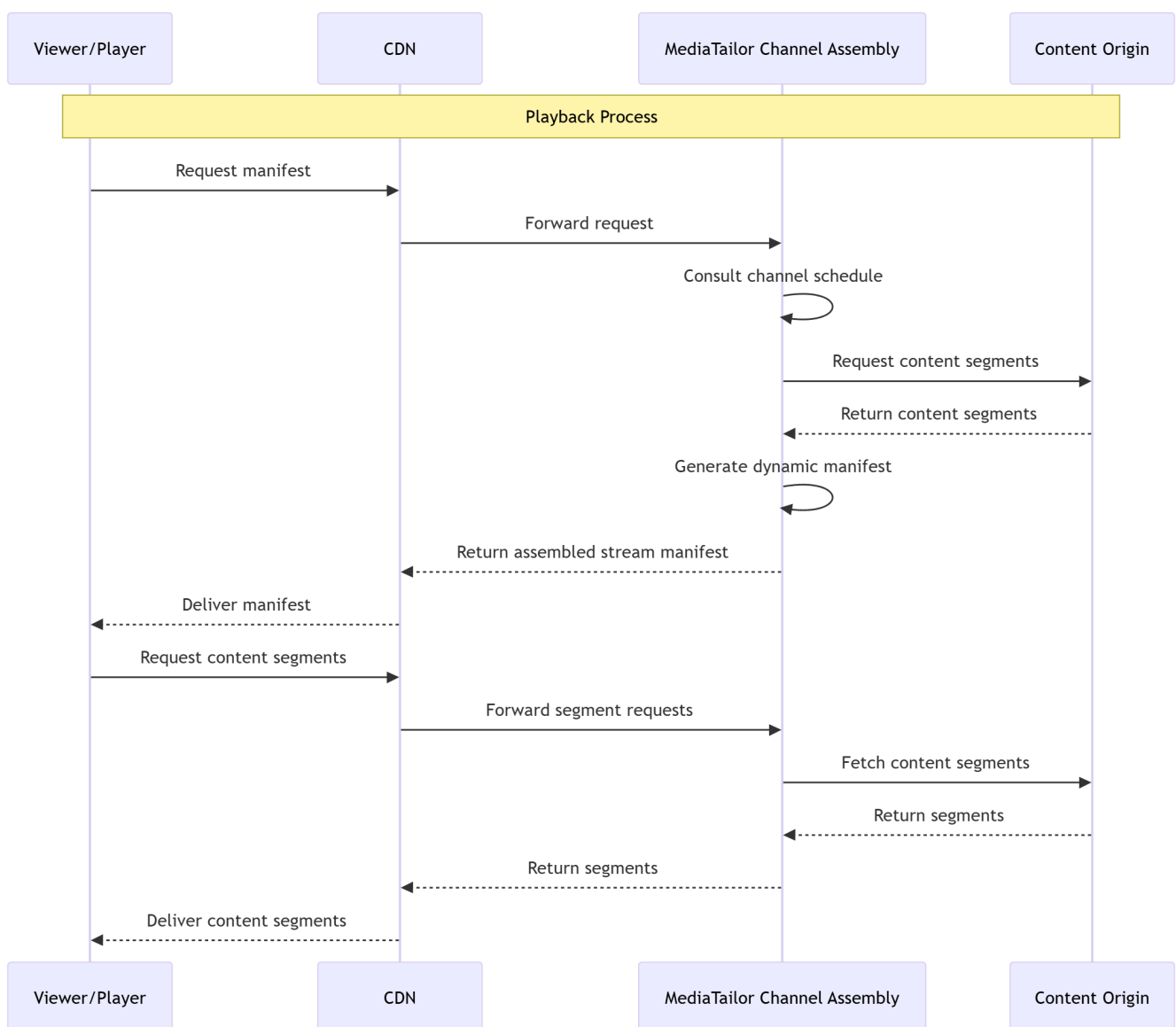
- [Entenda a arquitetura CDN de montagem de MediaTailor canais](#)
- [Configure a montagem básica MediaTailor do canal com uma CDN](#)
- [Configurar base URLs para montagem de MediaTailor canais CDN](#)
- [Implemente a inserção de MediaTailor anúncios com montagem de canais](#)
- [Configure a visualização com mudança de horário para montagem de canais MediaTailor](#)
- [Monitore as operações de CDN de montagem de MediaTailor canais](#)
- [Otimização completa de CDN de montagem de canais](#)

Entenda a arquitetura CDN de montagem de MediaTailor canais

AWS Elemental MediaTailor a montagem de canais se integra às redes de distribuição de conteúdo (CDNs) para fornecer canais de streaming linear com melhor desempenho e alcance global. A arquitetura recomendada posiciona a CDN entre os espectadores e a montagem do canal, com a montagem do canal acessando o conteúdo diretamente de sua origem. Este tópico explica os principais componentes da arquitetura e como eles funcionam juntos para fornecer seu conteúdo.

1. Os espectadores solicitam manifestos da CDN
2. O CDN encaminha solicitações para a montagem do canal
3. A montagem do canal reúne os manifestos de fontes de VOD
4. A montagem do canal retorna os manifestos para a CDN, que os encaminha aos telespectadores
5. Os espectadores solicitam segmentos por meio da CDN
6. A CDN encaminha as solicitações de segmento para a origem apropriada

Essa arquitetura garante um desempenho ideal enquanto mantém os benefícios de segurança e flexibilidade do uso de uma CDN.



Terminologia CDN para montagem de canais

A compreensão desses termos-chave ajudará você a implementar e solucionar problemas de integração de CDN de montagem de canais:

CDN de origem e CDN de borda

CDN de origem: uma CDN posicionada entre MediaTailor e a origem do seu conteúdo. Ele armazena segmentos de conteúdo em cache para reduzir a carga em seus servidores de origem. Em uma arquitetura multi-CDN, essa é a primeira camada CDN que interage diretamente com a origem.

Edge CDN: Uma CDN posicionada entre visualizadores e MediaTailor. Ele oferece manifestos e conteúdo personalizados aos espectadores. Em uma arquitetura multi-CDN, essa é a camada CDN mais externa que interage diretamente com os visualizadores.

Termos de configuração do CDN

Comportamento do cache: regras que determinam como uma CDN lida com diferentes tipos de solicitações, incluindo a duração do cache e o roteamento de origem.

TTL (Time To Live): a duração pela qual o conteúdo permanece válido em um cache CDN antes de precisar ser atualizado a partir da origem. Para obter recomendações detalhadas de TTL, consulte [Otimização de cache para CDN e integrações MediaTailor](#).

Chave de cache: o identificador exclusivo usado por uma CDN para armazenar e recuperar conteúdo em cache, geralmente incluindo caminho de URL, parâmetros de consulta e cabeçalhos.

Escudo de origem: uma camada intermediária de cache entre os pontos de presença da CDN e seu servidor de origem que reduz o número de solicitações para sua origem.

Colapso da solicitação: um recurso de CDN que combina várias solicitações simultâneas para o mesmo conteúdo em uma única solicitação de origem.

MediaTailor-termos específicos da CDN

Prefixo do segmento de conteúdo CDN: O nome de domínio CDN MediaTailor usado na geração de segmentos de conteúdo em URLs manifestos.

Prefixo do segmento de anúncio CDN: o nome de domínio CDN MediaTailor usado na geração de segmentos de anúncios em URLs manifestos.

Para obter mais informações sobre a configuração de CDN com MediaTailor, consulte [Configurar a integração com a CDN](#).

Configure a montagem básica MediaTailor do canal com uma CDN

AWS Elemental MediaTailor a montagem de canais permite que você configure uma integração básica com sua rede de distribuição de conteúdo (CDN) para uma entrega eficiente de canais de streaming linear para seus espectadores. Siga estas etapas para configurar a integração entre a montagem do canal e sua CDN.

1. Configure sua CDN para aceitar manifestos dos espectadores e encaminhá-los para a montagem do MediaTailor canal.
2. Configure a montagem MediaTailor do canal para acessar a programação do seu canal e determinar a programação atual.
3. Configure a montagem do MediaTailor canal para solicitar segmentos de conteúdo do seu servidor de origem com base na programação.
4. Garanta que sua origem de conteúdo possa entregar os segmentos solicitados para a montagem MediaTailor do canal.
5. Configure a montagem MediaTailor do canal para gerar manifestos dinâmicos com base na programação atual.
6. Configure sua CDN para fornecer as playlists multivariantes reunidas, as playlists de mídia e para os espectadores. MPDs
7. Configure sua CDN para lidar com solicitações de segmentos dos visualizadores, com as configurações de cache apropriadas.
8. Configure sua CDN para encaminhar as falhas de cache para a montagem MediaTailor do canal.
9. Configure a montagem de MediaTailor canais para recuperar os segmentos solicitados da sua origem de conteúdo.
10. Configure sua CDN para entregar segmentos de conteúdo aos espectadores para reprodução.

Configurar base URLs para montagem de MediaTailor canais CDN

AWS Elemental MediaTailor a montagem de canais requer uma configuração adequada de URL base para garantir que o roteamento de conteúdo pela sua rede de distribuição de conteúdo (CDN) funcione corretamente. Defina as configurações de URL base na montagem do canal para permitir a entrega bem-sucedida de conteúdo aos espectadores.

Configuração de URL do segmento de conteúdo

Na configuração do canal de montagem de canais, defina o URL base para seu domínio CDN. Isso garante que todos os segmentos URLs no manifesto montado apontem para sua CDN em vez de diretamente para seu servidor de origem.

Por exemplo, se seu conteúdo de origem estiver em `http://origin.example.com/content/` e seu domínio CDN estiver `https://cdn.example.com/`, defina o URL base como `https://cdn.example.com/content/`

Configuração de restrição de acesso

Para aumentar a segurança, configure sua CDN para restringir o acesso direto ao seu servidor de origem:

1. Configure os controles de acesso de origem em sua CDN.
2. Configure seu servidor de origem para aceitar somente solicitações do seu CDN.
3. Use cookies assinados URLs ou para autenticação do visualizador, se necessário.

Para a Amazon CloudFront, você pode usar o Origin Access Control (OAC) para proteger o acesso à sua origem. Para obter mais informações sobre como proteger sua CloudFront integração, consulte [Integração do CloudFront](#).

Implemente a inserção de MediaTailor anúncios com montagem de canais

A montagem de canais AWS Elemental MediaTailor se integra perfeitamente à inserção de anúncios do lado do servidor (SSAI) e às redes de entrega de conteúdo (CDNs) para criar canais lineares monetizados com publicidade personalizada.

Ao combinar a montagem de canais com a SSAI, você pode criar canais lineares que oferecem anúncios personalizados aos espectadores e, ao mesmo tempo, mantêm experiências com qualidade de transmissão. Essa integração permite que você:

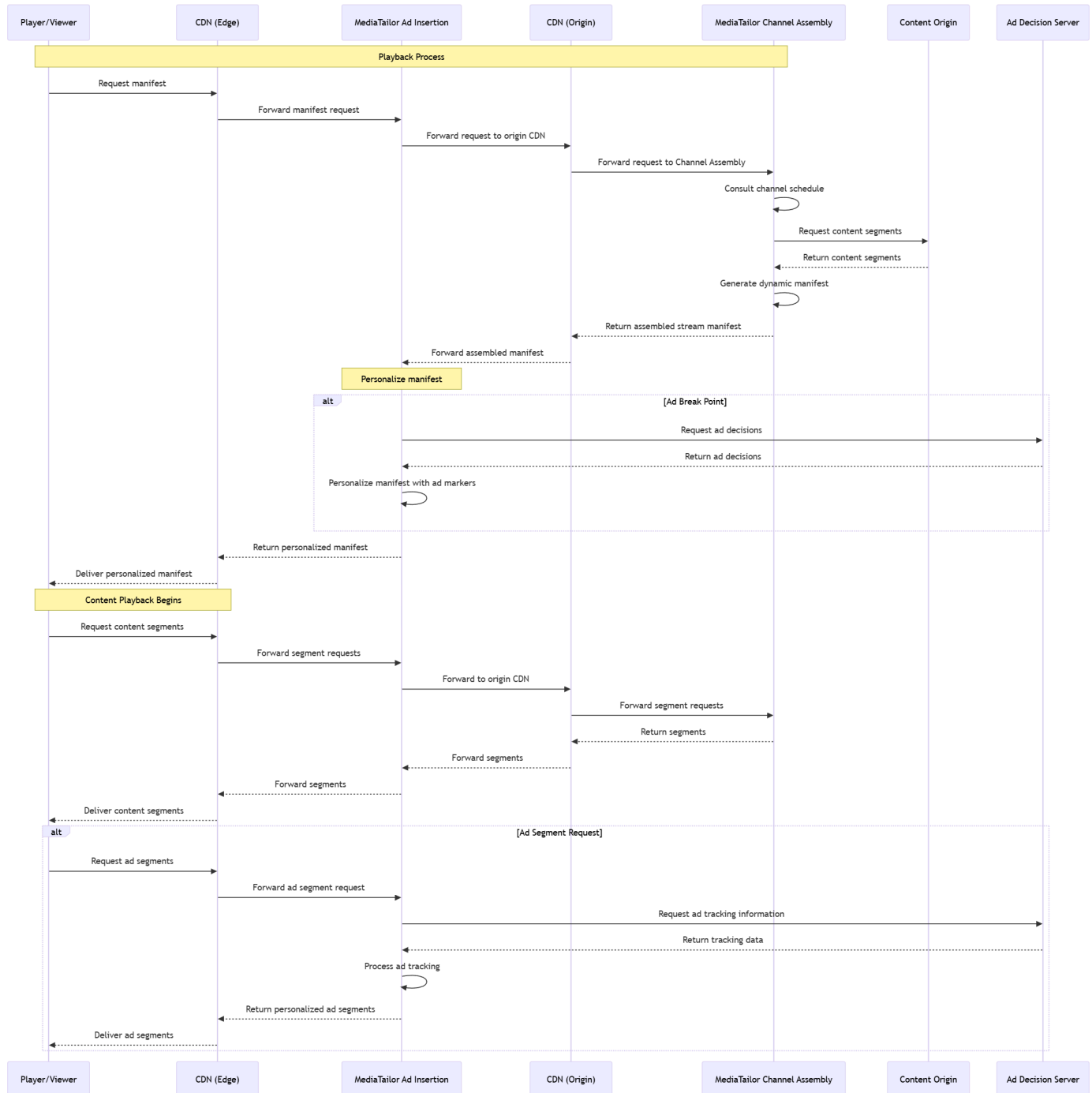
- Monetize o conteúdo - Gere receita por meio de publicidade direcionada em seus canais lineares
- Personalize experiências: exiba anúncios diferentes para os espectadores que assistem ao mesmo canal com base em seus perfis
- Mantenha a qualidade - garanta transições perfeitas entre conteúdo e anúncios para uma visualização com qualidade de transmissão

- Dimensione com eficiência - Support milhões de visualizadores simultâneos por meio da entrega de CDN

Para obter informações detalhadas sobre o SSAI com CDNs, consulte [Inserção de anúncios com CDN](#).

1. Configure sua CDN de borda para aceitar solicitações de manifesto dos espectadores e encaminhá-las para a inserção de MediaTailor anúncios.
2. Configure a inserção de MediaTailor anúncios para encaminhar solicitações para sua CDN de origem.
3. Configure sua CDN de origem para encaminhar solicitações para a montagem MediaTailor do canal.
4. Configure a montagem MediaTailor do canal para gerar manifestos dinâmicos com base na programação atual.
5. Configure sua CDN de origem para encaminhar os manifestos reunidos para a inserção de MediaTailor anúncios.
6. Configure a inserção de MediaTailor anúncios para solicitar decisões de anúncios do seu servidor de decisão de anúncios nos pontos de interrupção do anúncio.
7. Configure a inserção de MediaTailor anúncios para personalizar manifestos com marcadores de anúncios.
8. Configure sua CDN de ponta para oferecer manifestos personalizados aos espectadores.
9. Configure sua arquitetura de CDN para lidar com solicitações de conteúdo e segmento de anúncios de forma eficiente.

O diagrama a seguir ilustra esse fluxo de trabalho combinado:



Para um desempenho ideal ao combinar a montagem de canais e o SSAI:

- Configure comportamentos de cache que diferenciem entre montagem de canais e solicitações SSAI

- Defina valores TTL apropriados para manifestos e segmentos, conforme recomendado em [Otimização de cache para CDN e integrações MediaTailor](#)
- Garanta o roteamento adequado entre a montagem do canal, a inserção do anúncio e as origens da sua CDN
- Monitore as métricas de desempenho dos componentes de montagem de canais e inserção de anúncios

Para obter informações detalhadas sobre como configurar o SSAI com CDNs, consulte:

- [Entenda a arquitetura de inserção de anúncios para CDN e integrações MediaTailor](#)- Aprenda sobre a arquitetura e os conceitos da SSAI
- [Configure o MediaTailor SSAI básico com uma CDN para otimizar a entrega de anúncios](#)- Instruções Step-by-step de configuração do SSAI
- [Solucione problemas de MediaTailor SSAI com a entrega ininterrupta CDNs de anúncios](#)- Solucionar problemas comuns de integração do SSAI

Configure a visualização com mudança de horário para montagem de canais MediaTailor

AWS Elemental MediaTailor a montagem de canais oferece suporte a recursos de visualização com mudança de horário que permitem funcionalidades semelhantes às de DVR, como pausa, retrocesso e reinicialização de seus canais lineares. Ative esses recursos configurando sua rede de distribuição de conteúdo (CDN) para oferecer suporte à visualização com mudança de horário, o que permite que os espectadores controlem sua experiência de visualização.

Entendendo a visualização com mudança de horário

A visualização com mudança de tempo permite uma funcionalidade semelhante a um DVR para canais lineares, incluindo:

- **Início:** os espectadores podem começar a assistir a um programa desde o início, mesmo se entrarem depois de ele ter começado
- **Pausar e continuar:** os espectadores podem pausar o conteúdo e continuar assistindo mais tarde
- **Retroceder e avançar rapidamente:** os espectadores podem navegar para trás e para frente pelo conteúdo disponível

- Visualização atrasada: os espectadores podem assistir ao conteúdo que foi ao ar no início da programação do canal

A visualização com mudança de horário funciona adicionando um `start` parâmetro ao URL de reprodução do canal. O parâmetro especifica um deslocamento em segundos em relação à hora atual:

- Valores negativos indicam uma época no passado (como `start=-3600` significa “começar há 1 hora”)
- Valores positivos indicam uma hora no futuro (como `start=3600` significa “começar em 1 hora no futuro”)

Exemplo de URL com parâmetro de mudança de horário:

```
https://example-cdn.com/out/v1/channel-name/index.m3u8?start=-3600
```

Resolução de atraso de tempo

Para oferecer suporte à visualização com mudança de horário com uma CDN:

1. Configure sua CDN para encaminhar o parâmetro de `start` consulta para a montagem do canal.
2. Configure comportamentos de cache que incluam o `start` parâmetro na chave de cache.
3. Para manifestos com parâmetros de mudança de horário, use um TTL curto ou nenhum armazenamento em cache.

Isso garante que cada espectador receba o manifesto correto para a posição de horário solicitada.

Requisitos de CDN para mudança de horário

Sua CDN deve atender a esses requisitos para oferecer suporte à visualização com mudança de horário com montagem de canais:

- Encaminhe todos os parâmetros de consulta para a montagem do canal.
- Inclua o `start` parâmetro na chave de cache.
- Support a invalidação adequada do cache para manifestos com mudança de horário.
- Manipule respostas de manifesto variáveis com base nos parâmetros de consulta.

Monitore as operações de CDN de montagem de MediaTailor canais

AWS Elemental MediaTailor a montagem de canais requer monitoramento eficaz quando integrada a uma rede de distribuição de conteúdo (CDN) para garantir a entrega confiável de conteúdo. Implemente estratégias de monitoramento para a montagem do canal e a integração com a CDN para ajudar a garantir a entrega confiável de conteúdo e a rápida resolução de problemas.

Para obter uma orientação abrangente de monitoramento de CDN, incluindo métricas essenciais, configuração de ferramentas de monitoramento, configuração de alertas e estratégias de solução de problemas que se aplicam a todas as MediaTailor implementações, consulte [Monitoramento de CDN](#). Esta seção se concentra nos requisitos de monitoramento específicos da montagem do canal.

Implemente monitoramento específico para a montagem do seu canal e integração com a CDN:

- Acompanhe as métricas de geração de manifestos na montagem de canais.
- Monitore as solicitações de visualização com mudança de horário e seu impacto nas taxas de acerto do cache da CDN.
- Configure alertas para padrões incomuns em solicitações de manifesto.
- Implemente o rastreamento da disponibilidade do segmento em suas fontes de conteúdo.

[Para a Amazon CloudFront, crie um painel que integre métricas de CDN com MediaTailor métricas para visualizar todo o seu pipeline de entrega usando painéis. CloudWatch](#)

Se você também estiver usando o SSAI com a montagem do seu canal, consulte [Monitore as operações de CDN](#) para obter recomendações adicionais de monitoramento específicas para a inserção de anúncios.

Otimização completa de CDN de montagem de canais

Para obter orientações detalhadas de otimização de CDN específicas para implementações de montagem de canais, incluindo estratégias de armazenamento em cache, configuração de roteamento de solicitações, benchmarks de desempenho e técnicas avançadas de otimização, consulte [Otimização do desempenho da CDN](#).

O guia de otimização consolidado fornece:

- Configurações de cache específicas do conjunto de canais com valores TTL recomendados
- Otimização de roteamento de solicitações para entrega de manifestos e segmentos

- Benchmarks e metas de desempenho para todas as MediaTailor implementações
- Técnicas avançadas de otimização, incluindo Origin Shield e compressão
- Orientação combinada do fluxo de trabalho para montagem de canais com SSAI

Integre MediaTailor com uma MediaPackage CDN

AWS Elemental MediaTailor se integra AWS Elemental MediaPackage para fornecer anúncios em vídeo personalizados por meio de uma rede de entrega de conteúdo (CDN). MediaPackage é um serviço de embalagem e originação de just-in-time vídeo que prepara e protege seu conteúdo de vídeo para entrega pela Internet. Ele pega seu conteúdo de vídeo ao vivo ou sob demanda e o empacota em formatos de streaming, como HLS e DASH, preparando-o para espectadores em vários dispositivos.

Ao combinar MediaPackage com MediaTailor uma CDN, você cria um fluxo de trabalho de streaming completo que oferece anúncios personalizados em grande escala. A CDN distribui seu conteúdo globalmente, reduzindo a latência e melhorando a experiência do espectador, enquanto MediaTailor insere anúncios direcionados em seus streams.

Este tópico se concentra nas etapas essenciais de integração para MediaTailor que sua CDN e sua CDN trabalhem juntos. MediaPackage Para opções avançadas de configuração, solução de problemas e orientação de monitoramento, consulte [Próximas etapas](#).

Entendendo o fluxo de MediaPackage trabalho e o CDN

Antes de configurar sua integração, é importante entender como MediaPackage MediaTailor, e sua CDN trabalham juntas:

1. Preparação de conteúdo: MediaPackage recebe seu conteúdo de vídeo ao vivo ou sob demanda e o empacota em formatos de streaming (manifestos e segmentos HLS ou DASH).
2. Inserção de anúncios: MediaTailor solicita manifestos de MediaPackage, insere anúncios personalizados e veicula os manifestos modificados aos espectadores.
3. Distribuição global: sua CDN armazena em cache e distribui os segmentos de conteúdo (de MediaPackage) e os segmentos de anúncios (de MediaTailor) para espectadores em todo o mundo.
4. Reprodução do visualizador: os reprodutores de vídeo solicitam manifestos por meio da CDN, que roteia as solicitações adequadamente entre MediaTailor (para manifestos) e MediaPackage (para segmentos de conteúdo).

Essa arquitetura oferece vários benefícios:

- Escalabilidade: o CDN lida com altas cargas de visualização sem afetar seus servidores de origem
- Desempenho: o conteúdo é entregue a partir de locais periféricos mais próximos dos espectadores
- Eficiência de custos: custos reduzidos de largura de banda por meio do armazenamento em cache
- Confiabilidade: vários pontos de presença fornecem redundância

Pré-requisitos

Antes de começar, verifique se você tem os seguintes componentes configurados:

1. MediaPackage endpoint: um MediaPackage endpoint configurado que está recebendo e empacotando seu conteúdo de vídeo. Para obter instruções de configuração, consulte [Introdução MediaPackage](#) no guia do MediaPackage usuário.
2. MediaTailor configuração: uma MediaTailor configuração que usa seu MediaPackage endpoint como origem do conteúdo. Para obter instruções de configuração, consulte [Integrando uma fonte de conteúdo para inserção de MediaTailor anúncios](#).
3. Distribuição CDN: uma distribuição CDN (como CloudFront) configurada para funcionar com mídia de streaming. Para obter instruções de configuração, consulte [Criação de uma distribuição](#) no guia do CloudFront desenvolvedor.
4. Servidor de decisão de anúncios: um servidor de decisão de anúncios configurado que retorna respostas VAST ou VMAP para inserção de anúncios.

Etapa 1: definir as configurações essenciais de CDN

A configuração adequada da CDN é fundamental para uma MediaPackage integração bem-sucedida. Configurações incorretas podem causar falhas na reprodução, baixo desempenho do cache e aumento de custos. Sem as políticas de cache e o encaminhamento de parâmetros de consulta corretos, sua CDN pode não entregar os manifestos corretamente ou ignorar completamente o armazenamento em cache, levando a uma alta carga do servidor de origem e à degradação da experiência do visualizador.

Definir configurações básicas de cache

Configurar o cache básico é essencial porque MediaPackage usa cabeçalhos específicos de controle de cache para otimizar a entrega de conteúdo. Sem as configurações de cache adequadas,

sua CDN pode ignorar esses cabeçalhos, levando a solicitações de origem desnecessárias e aumentando a latência. Siga estas etapas para garantir o melhor comportamento de armazenamento em cache:

Para configurar o cache básico que funciona com MediaPackage:

1. Abra suas configurações de CloudFront distribuição no CloudFront console.
2. Selecione ou crie uma política de cache para sua MediaPackage origem.
3. Ative a opção “Cabeçalhos de controle de cache de origem”.
4. Permita MediaPackage controlar o comportamento do cache por meio de seus cabeçalhos de controle de cache.

Essa configuração básica permite MediaPackage definir automaticamente as durações de cache apropriadas para diferentes tipos de conteúdo. Para implementar a otimização avançada do cache com valores específicos de TTL e ajuste de desempenho, conclua primeiro essa configuração básica e continue [Otimize o armazenamento em cache da CDN MediaTailor e a entrega de conteúdo MediaPackage](#).

Configurar parâmetros de consulta essenciais

A configuração dos parâmetros de consulta é crucial para MediaPackage a funcionalidade. Sua CDN deve encaminhar parâmetros de consulta específicos para ativar recursos como visualização com mudança de horário e streaming de baixa latência. Configurações incorretas dos parâmetros de consulta podem impedir que esses recursos funcionem e estou preocupado em reduzir a eficiência do cache. Siga estas etapas para configurar o encaminhamento de parâmetros de consulta:

Para garantir que sua CDN encaminhe os parâmetros de consulta necessários para: MediaPackage

1. Nas configurações CloudFront de distribuição, selecione ou crie uma política de cache para solicitações de manifesto.
2. Em “Configurações da chave de cache”, selecione “Incluir cadeias de caracteres de consulta especificadas”.
3. Adicione os seguintes parâmetros de consulta essenciais:
 - `start&end` - Para funcionalidade de visualização com mudança de horário. Esses parâmetros são transmitidos MediaPackage para definir janelas de conteúdo específicas para visualização inicial e atualização.
 - `_HLS_msne _HLS_part` - Para suportar solicitações de reprodução LL-HLS

- `m-` Para capturar a modificação `ti/compame` do endpoint. MediaPackage as respostas sempre incluem a `?m=###` tag para capturar a hora modificada do endpoint. Se o conteúdo já estiver armazenado em cache com um valor diferente para essa tag, CloudFront solicita um novo manifesto em vez de servir a versão em cache
 - `aws.manifestfilter-` Para funcionalidade de filtragem de manifestos. Se você estiver usando a filtragem de manifesto, esse parâmetro deverá ser incluído para configurar a distribuição para encaminhar a string de `aws.manifestfilter` consulta para a MediaPackage origem, o que é necessário para que o recurso de filtragem de manifestos funcione.
4. Inclua somente as cadeias de caracteres de consulta que MediaPackage usa. A inclusão de cadeias de caracteres de consulta desnecessárias reduz a eficiência do cache ao criar várias variações de cache para o mesmo conteúdo.

Esses parâmetros habilitam a MediaPackage funcionalidade básica com sua CDN. Se você precisar implementar a filtragem de conteúdo para diferentes níveis de assinatura ou tipos de dispositivos, primeiro conclua essa configuração básica de parâmetros de consulta e prossiga para. [Configurar a filtragem de manifestos com MediaTailor MediaPackage, e CDN](#)

Para obter informações sobre como MediaTailor passa parâmetros de consulta, como `start` e `end` MediaPackage para visualização com mudança de horário, consulte [MediaTailor tratamento de parâmetros de consulta para origens](#) em. [MediaTailor parâmetros de consulta do manifesto](#)

Configurar o tempo limite de resposta para LL-HLS

A configuração de tempo limite é fundamental para o HLS de baixa latência porque o LL-HLS usa um mecanismo de “solicitações de bloqueio” em que o CDN espera por novos segmentos de conteúdo. Se o tempo limite for muito curto, as solicitações falharão antes que MediaPackage possam responder com novos segmentos, causando interrupções na reprodução e uma experiência ruim para o espectador. Configure os tempos limite apropriados para garantir uma reprodução suave de LL-HLS:

Se você estiver usando HLS de baixa latência, defina suas configurações de tempo limite de CDN:

1. Nas configurações de CDN, localize a configuração de tempo limite de origem.
2. Defina o valor do tempo limite de resposta para pelo menos três vezes a duração das peças.
3. Por exemplo, se a duração das peças for de 0,3 segundos, defina o tempo limite para pelo menos 0,9 segundos.

Isso garante que o CDN espere o tempo suficiente para responder MediaPackage ao usar o mecanismo de solicitações de bloqueio.

Etapa 2: verifique sua integração

Testar sua integração é essencial para garantir que todos os componentes funcionem juntos corretamente antes que seus espectadores tenham problemas. Uma falha na integração pode resultar em reprodução interrompida, anúncios perdidos ou baixo desempenho. Esse processo de verificação ajuda você a identificar e resolver problemas em um ambiente controlado.

Depois de definir suas configurações de CDN, verifique se sua integração está funcionando corretamente testando o fluxo de trabalho completo, desde a solicitação de conteúdo até a inserção do anúncio.

Etapa 2.1: testar a reprodução básica

O teste básico de reprodução verifica se sua CDN trata corretamente as solicitações de manifesto e as encaminha para o MediaTailor. Esse teste ajuda a identificar problemas de configuração com políticas de cache, encaminhamento de parâmetros de consulta e tratamento de manifestos. Siga estas etapas para testar a entrega básica do manifesto:

Teste se sua integração básica está funcionando solicitando um manifesto por meio de sua CDN:

1. Use um navegador da web ou curl para solicitar uma URL de manifesto por meio de sua CDN.
2. Verifique se o manifesto foi carregado com sucesso e se contém conteúdo e segmentos de anúncios.
3. Verifique esse segmento de conteúdo URLs no ponto de manifesto para seu domínio CDN.
4. Confirme se o segmento de anúncios URLs também aponta para seu domínio CDN.

Se o manifesto for carregado corretamente e contiver o esperado URLs, sua integração básica está funcionando. Para metodologias de teste abrangentes e procedimentos avançados de validação, consulte [Teste e validação para CDN e integrações MediaTailor](#). Para configurar um monitoramento abrangente do desempenho e da integridade de sua integração, consulte [Monitore o desempenho MediaPackage, CDN e integrações MediaTailor](#).

Etapa 2.2: Teste a reprodução de vídeo

O teste de reprodução de vídeo garante que sua integração completa funcione end-to-end, incluindo inserção de anúncios e entrega de conteúdo por meio de sua CDN. Esse teste verifica se os

segmentos de conteúdo e os segmentos de anúncios estão devidamente armazenados em cache e entregues, e se a experiência do espectador atende aos seus padrões de qualidade. Siga estas etapas para testar a funcionalidade de reprodução completa:

Teste se a reprodução de vídeo funciona corretamente com os anúncios inseridos:

1. Use um player de vídeo (como Video.js ou HLS.js) para reproduzir seu conteúdo por meio da CDN.
2. Verifique se o vídeo é reproduzido sem problemas de buffer.
3. Confirme se os anúncios são inseridos nos horários esperados durante a reprodução.
4. Verifique se o conteúdo e os segmentos de anúncios são carregados da sua CDN (não diretamente das origens).

Se a reprodução funcionar sem problemas com anúncios, sua integração está funcionando corretamente. Para metodologias de teste abrangentes e procedimentos avançados de validação, consulte [Teste e validação para CDN e integrações MediaTailor](#). Se você tiver problemas de reprodução, armazenamento em buffer ou inserção de anúncios, consulte. [Solução de problemas MediaPackage, CDN e integrações MediaTailor](#)

Próximas etapas

Depois de concluir a integração básica, você pode implementar recursos e otimizações avançados:

Otimização avançada de CDN

Para obter otimização detalhada do cache, configuração de TTL e ajuste de desempenho, consulte [Otimize o armazenamento em cache da CDN MediaTailor e a entrega de conteúdo MediaPackage](#).

Filtragem de manifestos

Para implementar a filtragem de conteúdo para serviços em camadas, otimização de dispositivos ou controle de acesso, consulte. [Configurar a filtragem de manifestos com MediaTailor MediaPackage, e CDN](#)

Solução de problemas

Se você encontrar problemas com sua integração, consulte [Solução de problemas MediaPackage, CDN e integrações MediaTailor](#).

Monitoramento do desempenho

Para configurar um monitoramento abrangente e entender as principais métricas de desempenho, consulte [Monitore o desempenho MediaPackage, CDN e integrações MediaTailor](#).

Configurar a filtragem de manifestos com MediaTailor MediaPackage, e CDN

AWS Elemental MediaTailor usa a filtragem de manifestos AWS Elemental MediaPackage para personalizar quais streams de áudio e vídeo são incluídos nos manifestos entregues a diferentes espectadores por meio de uma rede de distribuição de conteúdo (CDN). Isso é particularmente útil para implementar ofertas de serviços em camadas, otimizações específicas de dispositivos ou controles de acesso ao conteúdo.

Este tópico se concentra especificamente na implementação de recursos de filtragem de manifestos. Antes de implementar a filtragem de manifestos, você deve concluir a configuração básica da integração da rede de distribuição de conteúdo. Se você ainda não configurou sua integração básica MediaPackage com a rede de distribuição de conteúdo, comece com [Integre MediaTailor com uma MediaPackage CDN](#).

Capacidades de filtragem de manifestos

Antes de implementar a filtragem de manifestos, entenda o que você pode fazer com esse recurso:

Principais recursos de filtragem

A filtragem de manifestos fornece vários recursos importantes que ajudam você a controlar a entrega de conteúdo:

- Restrinja o acesso do espectador a conteúdo premium (como 4K HEVC)
- Segmente tipos de dispositivos específicos com fluxos apropriados
- Filtre o conteúdo com base em taxas de amostragem de áudio, idiomas ou codecs de vídeo
- Ofereça diferentes níveis de qualidade para diferentes assinantes

Casos de uso comuns

Esses casos de uso demonstram como a filtragem de manifestos pode atender aos requisitos comerciais específicos:

Níveis de assinatura

Ofereça aos assinantes básicos streams de baixa resolução e, ao mesmo tempo, forneça aos assinantes premium acesso a conteúdo em 4K

Exemplo: nível básico limitado a 720p, nível Premium chega a 4K

Otimização de dispositivos

Forneça automaticamente os fluxos apropriados com base nos recursos do dispositivo

Exemplo: dispositivos móveis obtêm taxas de bits mais baixas, dispositivos inteligentes TVs obtêm maior qualidade

Gerenciamento de largura de banda

Limite a qualidade do stream durante os períodos de pico de uso para gerenciar os custos da rede

Exemplo: reduza a taxa de bits máxima durante eventos de alto tráfego

Conteúdo regional

Ofereça diferentes idiomas de áudio ou variantes de conteúdo com base na localização do espectador

Exemplo: filtrar automaticamente as faixas de áudio do idioma local

Para obter mais informações sobre conceitos de filtragem de manifestos, consulte [Filtragem de manifestos](#) no guia do AWS Elemental MediaPackage usuário.

Configure sua CDN para filtragem de manifestos

A configuração da CDN para filtragem de manifestos é essencial porque sua CDN deve encaminhar o parâmetro de `aws.manifestfilter` consulta MediaPackage para que a filtragem funcione. Sem o encaminhamento adequado da sequência de caracteres de consulta, os parâmetros do filtro serão removidos pela CDN e todos os visualizadores receberão manifestos não filtrados, independentemente do nível de assinatura ou dos recursos do dispositivo. Essa configuração garante que sua lógica de filtragem alcance MediaPackage e funcione conforme o esperado.

Para habilitar a filtragem de manifestos por meio de sua CDN, você precisa configurar o encaminhamento da sequência de caracteres de consulta:

1. Na sua CloudFront distribuição, crie ou edite o comportamento do cache para solicitações de manifesto.
2. Para a política de cache, crie uma nova política ou edite uma existente.
3. Em Configurações da chave de cache, escolha “Incluir cadeias de caracteres de consulta especificadas”.
4. Adicione `aws.manifestfilter` à lista de cadeias de caracteres de consulta permitidas.
5. Se você também estiver usando outros MediaPackage recursos, adicione seus parâmetros de consulta:
 - `start_end` - Para visualização com mudança de horário
 - `time_delay` - Para funcionalidade de atraso de tempo
 - `_HLS_msne_HLS_part` - Para LL-HLS

Para obter mais informações sobre a criação de distribuições, consulte [Criar uma distribuição](#) no guia do CloudFront desenvolvedor da Amazon.

Implemente a filtragem do lado do cliente

A implementação do lado do cliente é onde você define como seus reprodutores de vídeo e aplicativos solicitam conteúdo filtrado. Essa configuração determina o conteúdo que cada espectador recebe com base no nível de assinatura, nos recursos do dispositivo ou em outros critérios. A implementação adequada garante que os espectadores recebam somente o conteúdo ao qual deveriam ter acesso, mantendo a eficiência ideal do cache da CDN.

Para implementar a filtragem de manifestos em seus reprodutores de vídeo e aplicativos:

Como funciona a filtragem

O processo de filtragem funciona da seguinte forma:

1. Seu player de vídeo ou aplicativo solicita um URL de manifesto que inclui parâmetros de filtro
2. O CDN encaminha a solicitação (incluindo parâmetros de consulta) para MediaTailor
3. MediaTailor passa os parâmetros do filtro para MediaPackage ao solicitar o manifesto de origem
4. MediaPackage aplica os filtros e retorna um manifesto personalizado contendo somente as variantes que correspondem aos seus critérios
5. MediaTailor processa o manifesto filtrado para inserção de anúncios e o retorna ao player

Formato de URL para filtragem

Compreender o formato correto do URL é fundamental para uma implementação bem-sucedida da filtragem. A formatação incorreta do URL fará com que os parâmetros de filtragem sejam ignorados ou causem erros de HTTP. A estrutura do URL deve incluir parâmetros de filtro como cadeias de caracteres de consulta para as quais sua CDN encaminha. MediaPackage Siga estas etapas para implementar a formatação de URL adequada:

Para implementar a filtragem de manifestos em seus reprodutores de vídeo:

1. Modifique a solicitação de manifesto do seu jogador URLs para incluir os parâmetros de filtro apropriados.
2. Use o seguinte formato de URL com os parâmetros de consulta:

```
https://CloudFront-Domain/v1/master/MediaTailor-Config/index.m3u8?  
aws.manifestfilter=video_codec:h264;audio_language:en-US
```

3. Quando seu player solicitar essa URL, MediaTailor passará esses parâmetros para MediaPackage, resultando em um manifesto filtrado.

Cenários comuns de filtragem

Use esses exemplos para implementar cenários comuns de filtragem:

Entrega de conteúdo específico para dispositivos

Para filtrar com base nos recursos do dispositivo, adicione esse parâmetro à sua solicitação de manifesto:

```
aws.manifestfilter=video_codec:h264;audio_sample_rate:0-44100
```

Este exemplo limita o conteúdo a vídeo e áudio H.264 com taxas de amostragem de até 44,1 kHz, adequadas para dispositivos móveis.

Restrição de conteúdo premium

Para limitar o acesso a conteúdo com alta taxa de bits, adicione esse parâmetro à sua solicitação de manifesto:

```
aws.manifestfilter=video_bitrate:0-9000000
```

Este exemplo restringe as taxas de bits de vídeo a 9 Mbps ou menos, adequadas para níveis básicos de assinatura.

Seleção de idioma

Para filtrar por idiomas de áudio específicos, adicione esse parâmetro à sua solicitação de manifesto:

```
aws.manifestfilter=audio_language:fr,en-US,de
```

Este exemplo inclui somente faixas de áudio em francês, inglês dos EUA e alemão.

Segmentação de resolução

Para filtrar por resoluções de vídeo específicas, adicione esse parâmetro à sua solicitação de manifesto:

```
aws.manifestfilter=video_height:240-360,720-1080
```

Este exemplo inclui streams de vídeo com alturas entre 240 a 360 pixels e 720 a 1080 pixels, excluindo resoluções de médio alcance.

Filtragem baseada em codec

Para filtrar codecs de vídeo específicos, adicione esse parâmetro à sua solicitação de manifesto:

```
aws.manifestfilter=video_codec:h264,h265
```

Este exemplo inclui somente streams de vídeo H.264 e H.265, excluindo outros codecs.

Considerações e limitações especiais

Para evitar problemas comuns ao implementar a filtragem de manifestos:

Limitações técnicas

- Para manifestos TS, use grupos de reprodução de áudio para evitar a remoção de fluxos de vídeo multiplexados com fluxos de áudio filtrados
- Nos manifestos TS e CMAF, a taxa de amostragem de áudio e a taxa de bits de vídeo não são facilmente visíveis no manifesto para verificação

- Parâmetros de solicitação anexados a playlists ou segmentos de mídia resultarão em um erro HTTP 400

Condições de erro

- Se a filtragem resultar em um manifesto vazio (nenhum fluxo atende aos critérios de filtro), MediaPackage retornará um erro HTTP 400
- Configurações de filtro conflitantes (filtros de endpoint + parâmetros de consulta) resultam em erros HTTP 404
- Sintaxe de filtro inválida ou tipos de filtro não suportados resultam em erros HTTP 400

Considerações sobre a performance

- Cada combinação exclusiva de filtros cria uma entrada de cache separada, potencialmente reduzindo a eficiência do cache
- Filtros complexos com muitos critérios podem afetar o desempenho da geração de manifestos
- Considere o uso de filtros em nível de endpoint em cenários de filtragem estática para melhorar o desempenho do cache

Teste sua implementação de filtragem

Testar a implementação da filtragem de manifestos é fundamental para garantir que os espectadores recebam o conteúdo correto com base no nível de acesso e nos recursos do dispositivo. A falha na filtragem pode fazer com que os espectadores recebam níveis de qualidade incorretos, formatos incompatíveis ou conteúdo ao qual não deveriam ter acesso. Testes abrangentes ajudam a identificar e resolver esses problemas antes que eles afetem seus espectadores.

Para verificar se a filtragem do manifesto está funcionando corretamente:

1. Solicite manifestos com diferentes parâmetros de filtro e verifique os resultados
2. Verifique se os manifestos filtrados contêm somente os fluxos esperados
3. Casos extremos de teste (resultados vazios, filtros inválidos) para garantir o tratamento adequado de erros
4. Verifique se sua CDN está encaminhando corretamente os parâmetros do filtro
5. Teste com diferentes dispositivos e players para garantir a compatibilidade

Para solucionar problemas de filtragem, consulte [Solucionar problemas de integração do MediaPackage CDN](#).

Se você encontrar erros HTTP 400, manifestos vazios ou parâmetros de filtragem que não funcionem conforme o esperado, consulte [Solução de problemas MediaPackage, CDN e integrações MediaTailor](#) para obter orientações específicas sobre solução de problemas de filtragem de manifestos.

Otimize o armazenamento em cache da CDN MediaTailor e a entrega de conteúdo MediaPackage

AWS Elemental MediaTailor requer uma configuração adequada de cache da rede de distribuição de conteúdo (CDN) para um desempenho ideal ao ser usado AWS Elemental MediaPackage como origem de conteúdo. MediaPackage fornece cabeçalhos específicos de controle de cache que informam à sua rede de distribuição de conteúdo por quanto tempo armazenar diferentes tipos de conteúdo em cache. Seguir essas recomendações garante uma reprodução suave e uma entrega eficiente de conteúdo.

Este tópico se concentra especificamente na otimização do comportamento de armazenamento em cache para maximizar o desempenho e minimizar os custos. Antes de implementar a otimização avançada do cache, certifique-se de ter concluído a configuração básica da integração da rede de distribuição de conteúdo. Se você ainda não configurou sua integração básica, comece com [Integre MediaTailor com uma MediaPackage CDN](#).

MediaPackage cabeçalhos de controle de cache

MediaPackage define valores TTL específicos para diferentes tipos de conteúdo para otimizar o comportamento do armazenamento em cache:

Listas de reprodução multivariantes (HLS e LL-HLS)

TTL: metade da duração dos segmentos de mídia

Motivo: essas playlists mudam à medida que novos segmentos são disponibilizados, por isso precisam de atualizações frequentes

Listas de reprodução de mídia (HLS normal)

TTL: metade da duração dos segmentos de mídia

Motivo: semelhantes às playlists multivariantes, elas são atualizadas à medida que o conteúdo progride

Listas de reprodução de mídia (LL-HLS)

TTL: 1 segundo

Motivo: o streaming de baixa latência requer atualizações muito frequentes

Segmentos de mídia TS e segmentos iniciais

TTL: 1209600 segundos (14 dias)

Motivo: os segmentos de mídia não mudam depois de criados, então eles podem ser armazenados em cache por longos períodos

Segmentos de mídia e segmentos de inicialização do CMAF

TTL: 1209600 segundos (14 dias)

Motivo: como os segmentos TS, eles são imutáveis depois de criados

Para obter recomendações abrangentes de TTL em todos os MediaTailor fluxos de trabalho e estratégias adicionais de otimização de cache, consulte [Otimização de cache para CDN e integrações MediaTailor](#)

Configurar políticas de cache da CDN

A configuração adequada da política de cache é essencial para otimizar o desempenho e a economia. Diferentes tipos de conteúdo (manifestos, segmentos, arquivos de inicialização) têm diferentes requisitos de cache. O uso de comportamentos de cache separados permite otimizar o armazenamento em cache para cada tipo de conteúdo, melhorando as taxas de acerto do cache e reduzindo a carga de origem. Sem políticas de cache adequadas, você pode enfrentar solicitações de origem desnecessárias, aumento de custos e baixo desempenho de reprodução.

Para honrar adequadamente os cabeçalhos de MediaPackage controle de cache e otimizar o armazenamento em cache:

1. Abra suas configurações de CloudFront distribuição no CloudFront console.
2. Crie comportamentos de cache separados para diferentes tipos de conteúdo:
 - Solicitações de manifesto (*.m3u8, *.mpd)

- Segmentos de mídia (*.ts, *.mp4, *.m4s)
 - Segmentos de inicialização
3. Para cada comportamento de cache, crie ou selecione uma política de cache com estas configurações:
- Ative a opção “Cabeçalhos de controle de cache de origem”
 - Defina a “Política de solicitação de origem” para encaminhar os cabeçalhos necessários
 - Configurar o encaminhamento da sequência de caracteres de consulta com base no tipo de conteúdo

Comportamento do cache manifesto

Para solicitações de manifesto (*.m3u8, *.mpd):

- Padrão de caminho: *.m3u8 e *.mpd
- Política de cache: cabeçalhos de controle de cache de origem Honor
- Cadeias de caracteres de consulta: encaminhe parâmetros específicos (consulte [Otimize o encaminhamento da sequência de caracteres de consulta](#))
- Cabeçalhos: encaminhe todos os cabeçalhos (para obter os requisitos mínimos, consulte [Cabeçalhos necessários para integração com MediaTailor CDN](#))

Comportamento do cache do segmento de mídia

Para segmentos de mídia (*.ts, *.mp4, *.m4s):

- Padrão de caminho: *.ts, *.mp4, *.m4s
- Política de cache: cabeçalhos de controle de cache Honor Origin (TTL de 14 dias)
- Cadeias de caracteres de consulta: nenhuma (os segmentos não usam parâmetros de consulta)
- Compressão: habilite para melhorar o desempenho de entrega

Otimize o encaminhamento da sequência de caracteres de consulta

A otimização da sequência de caracteres de consulta é fundamental para a eficiência do cache porque parâmetros de consulta desnecessários criam várias variações de cache para o mesmo conteúdo. Cada combinação exclusiva de parâmetros de consulta cria uma entrada de cache separada, o que reduz as taxas de acerto do cache e aumenta as solicitações de origem. Ao

encaminhar somente as cadeias de caracteres de consulta que MediaPackage realmente usa, você maximiza a eficiência do cache enquanto mantém a funcionalidade completa.

Configure sua CDN para encaminhar somente as cadeias de caracteres de consulta que MediaPackage usa, melhorando a eficiência do cache:

Cadeias de caracteres de consulta essenciais

`start` `end` - Para janelas de visualização com mudança de horário

`time_delay`- Para aplicar atraso de tempo no conteúdo do manifesto

`_HLS_msn`, `_HLS_m`, e `_HLS_part` - Para solicitações de reprodução LL-HLS

Cadeias de consulta específicas de recursos

`aws.manifestfilter`- Para [filtragem de manifestos](#)

Important

Não inclua nenhuma outra sequência de caracteres de consulta em sua chave de cache. MediaPackage ignora parâmetros não reconhecidos e incluí-los reduz a eficiência do cache ao criar variações de cache desnecessárias.

Técnicas de otimização de desempenho

Essas otimizações são configuradas em sua CDN (como CloudFront), não em ou. MediaPackage MediaTailor Implemente essas otimizações adicionais para maximizar o desempenho do cache:

Escudo de origem

O Origin Shield fornece uma camada adicional de cache entre os pontos de extremidade e MediaPackage os pontos finais da sua CDN. Isso reduz o número de solicitações que chegam aos seus MediaPackage endpoints, o que pode melhorar o desempenho e reduzir os custos, especialmente durante picos de tráfego ou quando as taxas de acerto do cache estão abaixo do ideal. O Origin Shield é particularmente benéfico para transmissão ao vivo, onde vários pontos de presença podem solicitar o mesmo conteúdo simultaneamente.

Ative o Origin Shield para reduzir a carga em seus MediaPackage endpoints:

1. Na sua CloudFront distribuição, habilite o Origin Shield para sua MediaPackage origem.
2. Selecione uma região de escudo de origem próxima ao seu MediaPackage endpoint.
3. Isso cria uma camada de cache adicional que reduz as solicitações para MediaPackage.

Configuração de compressão

Ative a compactação para respostas baseadas em texto:

- Ativar compactação para arquivos de manifesto (*.m3u8, *.mpd)
- Não compacte segmentos de mídia (já compactados)
- Certifique-se de que todos os cabeçalhos sejam encaminhados para MediaPackage (para obter os requisitos mínimos, consulte) [Cabeçalhos necessários para integração com MediaTailor CDN](#)

Monitore o desempenho do cache

Acompanhe essas métricas principais para garantir o desempenho ideal do cache:

Taxa de acertos do cache

Meta: 90% ou mais para segmentos de mídia, 70% ou mais para manifestos

Taxas baixas podem indicar configurações de TTL incorretas ou parâmetros de consulta desnecessários

Volume de solicitações de origem

Monitore as solicitações que chegam aos MediaPackage endpoints

Volumes altos podem indicar problemas de cache

Variações da chave de cache

Revise os padrões das chaves de cache para identificar variações desnecessárias

Muitas variações reduzem a eficiência do cache

Depois de implementar essas otimizações de cache, configure o monitoramento para monitorar sua eficácia. Para obter orientação sobre como monitorar as taxas de acerto do cache, os padrões de solicitação de origem e outras métricas importantes de desempenho, consulte [Monitore o desempenho MediaPackage, CDN e integrações MediaTailor](#). Se você observar um desempenho

ruim do cache ou solicitações de origem inesperadas, consulte as etapas [Solução de problemas MediaPackage, CDN e integrações MediaTailor](#) de solução de problemas.

Monitore o desempenho MediaPackage, CDN e integrações MediaTailor

AWS Elemental MediaTailor requer monitoramento eficaz para manter o desempenho ideal de sua integração com AWS Elemental MediaPackage a rede de distribuição de conteúdo (CDN). Este tópico fornece orientação sobre as principais métricas a serem monitoradas, as ferramentas de monitoramento a serem usadas e como configurar alertas para a detecção proativa de problemas.

Antes de configurar o monitoramento, certifique-se de que sua integração básica esteja funcionando corretamente. Se você ainda não concluiu sua configuração básica de integração de rede de distribuição de conteúdo, comece com [Integre MediaTailor com uma MediaPackage CDN](#). Se você precisar solucionar problemas identificados por meio do monitoramento, consulte.. [Solução de problemas de integração de CDN](#)

Principais métricas de desempenho

Monitore essas métricas essenciais para garantir o desempenho ideal de sua MediaPackage integração com a CDN:

Métricas de desempenho da CDN

Para obter métricas abrangentes de desempenho da CDN, incluindo metas de taxa de acerto do cache, monitoramento do volume de solicitações de origem e benchmarks de tempo de resposta, consulte [Benchmarks de desempenho para CDN e integrações MediaTailor](#) o guia de otimização da CDN.

Principais considerações específicas do EMP para métricas de CDN:

Cabeçalhos de controle de cache EMP

O que verificar: garanta que sua CDN respeite os cabeçalhos de controle de cache do EMP para um comportamento TTL ideal

Comportamento esperado: diferentes tipos de conteúdo devem ter durações de cache diferentes com base nos cabeçalhos do EMP

Para obter orientações detalhadas sobre a otimização do cache do EMP, consulte [Otimizar o armazenamento em cache da CDN MediaTailor e a entrega de conteúdo MediaPackage](#).

Impacto do parâmetro de consulta

O que monitorar: acompanhe como os parâmetros de consulta específicos do EMP afetam a eficiência do cache

Alvo de otimização: garantir que somente os parâmetros de consulta EMP necessários sejam incluídos nas chaves de cache

Tempos de resposta da CDN

O que monitorar: acompanhe os tempos de resposta de diferentes tipos de conteúdo (manifestos versus segmentos).

Valores alvo:

- Conteúdo em cache: menos de 100 ms
- Solicitações de origem: menos de 500 ms

MediaPackage métricas de desempenho

Taxas de erro

O que monitorar: monitore as taxas de erro HTTP tanto do CDN quanto dos MediaPackage endpoints. Preste atenção especial aos erros 4xx, que podem indicar problemas de configuração.

Códigos de erro principais:

- 400 erros: geralmente relacionados a problemas de filtragem de manifestos
- Erros 404: podem indicar problemas de roteamento ou configuração
- Erros 504: problemas de tempo limite, especialmente com LL-HLS

Volume e padrões de solicitação

O que monitorar: acompanhe os padrões de solicitação nos MediaPackage endpoints para identificar tendências de uso e necessidades de capacidade.

Padrões para observar:

- Horários de pico de uso
- Distribuição geográfica das solicitações
- Distribuição do tipo de conteúdo (ao vivo versus sob demanda)

Métricas de latência

End-to-end latência

O que monitorar: Para implementações de LL-HLS, monitore a end-to-end latência desde a ingestão do conteúdo até a reprodução do visualizador. A alta latência pode indicar problemas de configuração da CDN.

Valores alvo:

- LL-HLS: menos de 3 segundos de latência glass-to-glass
- HLS normal: menos de 30 segundos

Tempo de geração do manifesto

O que monitorar: Tempo gasto MediaPackage para gerar manifestos, especialmente com a filtragem aplicada.

Valores alvo: menos de 200 ms para geração de manifesto

Ferramentas e configuração de monitoramento

Configurar ferramentas de monitoramento abrangentes é essencial para manter o desempenho ideal e identificar rapidamente os problemas antes que eles afetem os espectadores. Sem o monitoramento adequado, a degradação do desempenho, as ineficiências do cache ou os problemas de integração podem passar despercebidos até que os espectadores tenham uma qualidade de reprodução ruim. A configuração correta de monitoramento fornece visibilidade de todos os aspectos de sua MediaPackage integração com a CDN.

Use esses serviços e ferramentas da AWS para monitorar sua integração com MediaPackage a CDN:

Amazon CloudWatch

A Amazon CloudWatch fornece a base para monitorar sua MediaPackage integração com a CDN coletando e armazenando métricas de ambos os serviços. A CloudWatch configuração adequada garante que você tenha os dados necessários para identificar tendências de desempenho, solucionar problemas e otimizar sua integração. Sem CloudWatch métricas, você não terá visibilidade do desempenho do sistema e talvez não detecte problemas até que eles se tornem críticos.

Configure o CloudWatch monitoramento para uma coleta abrangente de métricas:

1. **MediaPackage métricas:** habilite CloudWatch métricas para seus MediaPackage endpoints para rastrear volumes de solicitações, taxas de erro e tempos de resposta.
2. **Métricas de CDN:** configure CloudWatch para coletar CloudFront métricas, incluindo taxas de acertos de cache, contagens de solicitações de origem e taxas de erro.
3. **Métricas personalizadas:** crie métricas personalizadas para negócios específicos, KPIs como engajamento do espectador ou popularidade do conteúdo.

CloudWatch painéis

Crie painéis abrangentes para visualizar suas métricas:

1. **Painel de visão geral:** métricas de alto nível que mostram a integridade geral do sistema
2. **Painel de desempenho da CDN:** métricas detalhadas da CDN, incluindo desempenho do cache e distribuição geográfica
3. **MediaPackage painel de desempenho:** métricas MediaPackage específicas, incluindo padrões de solicitação e taxas de erro
4. **Painel de latência:** métricas de End-to-end latência para diferentes tipos de conteúdo e regiões

Análise de registros

Configure a análise de log para solucionar problemas detalhados:

1. **Registros de acesso à CDN:** habilite e analise os registros de acesso à CDN para entender os padrões de solicitação e identificar problemas
2. **MediaPackage CloudWatch registros:** monitore MediaPackage os registros em busca de erros e problemas de desempenho
3. **Agregação de registros:** use o Amazon CloudWatch Logs Insights ou ferramentas de terceiros para analisar padrões de registros

Configurar alertas e notificações

A configuração de alertas é crucial para a detecção e resolução proativas de problemas. Sem os alertas adequados, os problemas podem passar despercebidos até afetarem significativamente a experiência do espectador ou causarem interrupções no serviço. Alertas bem configurados ajudam você a identificar e resolver problemas antes que eles afetem seus espectadores e garantem que sua equipe seja notificada sobre problemas críticos que exigem atenção imediata.

Configure alertas proativos para identificar problemas antes que eles afetem os espectadores:

Alertas críticos

Configure alertas imediatos para problemas críticos:

- Altas taxas de erro: alerta quando as taxas de erro de 4xx ou 5xx excederem 5% em um período de 5 minutos
- Diminuição da taxa de acertos do cache: alerta quando a taxa de acertos do cache cair abaixo de 70% para manifestos ou 85% para segmentos
- Alta latência: alerta quando a end-to-end latência exceder os limites desejados
- Picos de solicitações de origem: alerta quando as solicitações de origem aumentarem em mais de 50% em comparação com a linha de base

Alertas de aviso

Configure alertas de alerta para tendências que possam indicar problemas em desenvolvimento:

- Degradação gradual do desempenho: alerta quando os tempos de resposta aumentarem em 20% em um período de 30 minutos
- Tendências de eficiência do cache: alerta quando as taxas de acertos do cache mostrarem tendências decrescentes ao longo do tempo
- Padrões de tráfego incomuns: alerta para mudanças inesperadas no volume de solicitações ou na distribuição geográfica

Use dados de monitoramento para otimização

Aproveite os dados de monitoramento para melhorar continuamente o desempenho:

Análises regulares de desempenho

1. Revisões semanais: analise as tendências de desempenho semanais e identifique oportunidades de otimização
2. Planejamento mensal da capacidade: use padrões de tráfego para planejar as necessidades de capacidade e a otimização da CDN
3. Revisões trimestrais da arquitetura: avalie a eficiência geral da arquitetura e identifique áreas para melhoria

Ações comuns de otimização

Com base nos dados de monitoramento, considere estas ações de otimização:

- Ajustes da política de cache: modifique os valores de TTL com base nos padrões reais de atualização do conteúdo. Para obter orientações detalhadas sobre otimização de TTL, consulte [Otimização de cache para CDN e integrações MediaTailor](#).
- Otimização geográfica: adicione pontos de presença da CDN em regiões com alto tráfego
- Otimização de parâmetros de consulta: remova parâmetros de consulta desnecessários que fragmentam o cache
- Configuração do escudo de origem: implemente o escudo de origem em regiões com altos volumes de solicitações de origem

Para obter orientações detalhadas de monitoramento específicas para MediaPackage, consulte [Monitoramento MediaPackage](#) no guia MediaPackage do usuário.

Solução de problemas MediaPackage, CDN e integrações MediaTailor

AWS Elemental MediaTailor a integração com AWS Elemental MediaPackage a rede de distribuição de conteúdo (CDN) pode encontrar problemas comuns que afetam a reprodução, o armazenamento em cache ou outras funcionalidades de integração. Use este guia quando encontrar problemas de reprodução, de cache ou outros erros relacionados à integração.

Para obter uma orientação abrangente sobre solução de problemas de CDN, incluindo problemas de desempenho de cache universal, resolução de erros HTTP, procedimentos de teste e técnicas de diagnóstico que se aplicam a todas as MediaTailor implementações, consulte [Solucionar problemas de integração com a CDN](#). Esta seção se concentra nos requisitos MediaPackage específicos de solução de problemas.

Antes de solucionar problemas, verifique se você concluiu a configuração básica da integração corretamente. Se você ainda não configurou sua integração ou precisa revisar as etapas de configuração, consulte [Integre MediaTailor com uma MediaPackage CDN](#). Para obter orientação sobre como otimizar o desempenho do cache após a resolução de problemas, consulte [Cache CDN](#).

Erros de filtragem de manifestos

Problemas com a funcionalidade de filtragem de MediaPackage manifestos, com base em condições de erro documentadas:

Erros HTTP 400 com filtragem de manifesto

Sintomas: solicitações com `aws.manifestfilter` parâmetros retornam HTTP 400 Bad Request

Causas validadas (da AWS documentação):

- Os critérios de filtro resultam em um manifesto vazio (nenhum fluxo corresponde às condições do filtro)
- Nomes ou valores de parâmetros de filtro inválidos
- Sintaxe de sequência de caracteres de consulta malformada
- Parâmetros de filtro duplicados ou repetidos
- A sequência de parâmetros do filtro excede 1024 caracteres
- Parâmetros de consulta aplicados a playlists ou segmentos de mídia (não suportados)

Soluções:

1. Revise seus parâmetros de filtro para garantir que eles correspondam aos fluxos de conteúdo disponíveis. Se a filtragem não resultar em fluxos correspondentes, MediaPackage retornará HTTP 400.
2. Valide a sintaxe do filtro em relação aos nomes de parâmetros e formatos de valor suportados.
3. Verifique se há parâmetros duplicados em sua string de consulta.
4. Certifique-se de que os parâmetros do filtro sejam aplicados somente às playlists multivariantes, não às playlists ou segmentos de mídia.
5. Verifique se a sequência total de parâmetros tem menos de 1024 caracteres.

Referência: condições de [erro de filtragem de AWS Elemental MediaPackage manifestos](#)

A filtragem de manifestos não está funcionando (HTTP 200, mas nenhuma filtragem aplicada)

Sintomas: as solicitações retornam HTTP 200, mas o manifesto contém todos os fluxos em vez do subconjunto filtrado

Causas possíveis:

- CDN não encaminha o parâmetro de `aws.manifestfilter` consulta para MediaPackage
- Parâmetro de filtro não encontrado nos fluxos disponíveis (retorna o manifesto não filtrado com HTTP 200)

Soluções:

1. Verifique se sua política de cache de CDN está incluída `aws.manifestfilter` na lista de cadeias de caracteres de consulta encaminhadas.
2. Teste os parâmetros do filtro diretamente nos MediaPackage endpoints (ignorando a CDN) para verificar se eles funcionam conforme o esperado.
3. Verifique se os valores do filtro correspondem às características reais dos seus fluxos de conteúdo.

Referência: condições de [erro de filtragem de AWS Elemental MediaPackage manifestos](#)

Procedimentos de diagnóstico

Os procedimentos sistemáticos de diagnóstico ajudam você a identificar a causa raiz dos problemas de integração com rapidez e eficiência. Seguir uma abordagem estruturada evita perda de tempo com suposições incorretas e garante que você resolva o problema real em vez dos sintomas. Essas etapas de diagnóstico baseadas em evidências foram projetadas para isolar problemas e orientá-lo até a solução apropriada.

Siga estas etapas de diagnóstico baseadas em evidências para identificar problemas:

Analise o desempenho do cache

A análise do desempenho do cache é crucial para as integrações do EMP porque a baixa eficiência do cache leva ao aumento da carga de origem, a custos mais altos e a possíveis problemas de reprodução.

Para uma solução abrangente de problemas de desempenho de cache, incluindo análise da taxa de acertos de cache, otimização da chave de cache e etapas sistemáticas de diagnóstico, consulte [Problemas de desempenho do cache CDN](#) o guia principal de solução de problemas da CDN.

Considerações sobre cache específicas do EMP:

- Cabeçalhos de controle de cache do EMP: verifique se sua CDN respeita os cabeçalhos de controle de cache do EMP em vez de substituí-los
- Parâmetros de consulta EMP: Certifique-se de que somente os parâmetros de consulta EMP necessários sejam incluídos nas chaves de cache
- Comportamento do EMP TTL: confirme se os diferentes tipos de conteúdo do EMP têm durações de cache apropriadas

Para obter orientações detalhadas sobre como otimizar as políticas de cache do EMP e as configurações de TTL, consulte [Otimize o armazenamento em cache da CDN MediaTailor e a entrega de conteúdo MediaPackage](#)

Validar a configuração de filtragem de manifestos

A validação da filtragem de manifestos é essencial porque problemas de filtragem podem fazer com que os espectadores recebam conteúdo incorreto, formatos incompatíveis ou conteúdo ao qual eles não deveriam ter acesso. O teste sistemático ajuda a identificar se os problemas estão relacionados à configuração da CDN, à sintaxe dos parâmetros do filtro ou à disponibilidade do conteúdo.

Teste sistematicamente a funcionalidade de filtragem de manifestos:

1. Teste os parâmetros do filtro diretamente nos MediaPackage endpoints (ignorando a CDN) para verificar se eles funcionam corretamente.
2. Compare manifestos filtrados e não filtrados para confirmar se os fluxos esperados foram incluídos/excluídos.
3. Verifique se sua política de cache de CDN encaminha o parâmetro de `aws.manifestfilter` consulta.
4. Verifique se há erros de HTTP 400 e compare-os com as condições de erro documentadas.

Se você precisar implementar ou modificar a filtragem de manifestos após resolver problemas, consulte [Configurar a filtragem de manifestos com MediaTailor MediaPackage, e CDN](#) para obter orientações completas de configuração.

Validar a configuração dos parâmetros de consulta

Certifique-se de que sua CDN encaminhe somente os parâmetros de consulta necessários:

1. Revise sua política de cache de CDN para confirmar se ela inclui somente os parâmetros AWS recomendados:
 - `aws.manifestfilter`- para filtragem de manifestos
 - `aws.manifestsettings`- para visualização com mudança de horário
 - `_HLS_msne_HLS_part` - para suporte LL-HLS
2. Remova quaisquer outros parâmetros de consulta da sua chave de cache, pois MediaPackage os ignora e eles reduzem a eficiência do cache.

Referência: [Trabalhando com AWS Elemental MediaPackage e CDNs](#)

Referência de código de erro

Referência para condições de erro documentadas e suas causas:

Solicitação incorreta de HTTP 400 (filtragem de manifesto)

Causas documentadas:

- A aplicação dos resultados do filtro em um manifesto vazio
- Nomes ou valores de parâmetros inválidos
- Sintaxe de sequência de caracteres de consulta malformada
- Parâmetros de filtro duplicados
- A sequência de parâmetros excede 1024 caracteres
- Parâmetros de consulta em playlists ou segmentos de mídia

Referência: condições de [erro de filtragem de MediaPackage manifestos](#)

HTTP 200 OK (sem filtragem aplicada)

Causas documentadas:

- Parâmetro de filtro não encontrado nos fluxos disponíveis (retorna o manifesto não filtrado)
- Somente fluxos de legendas estão presentes após a filtragem (retorna o manifesto não filtrado)

Referência: condições de [erro de filtragem de MediaPackage manifestos](#)

Recursos adicionais para solução de problemas

Para questões não abordadas neste tópico, consulte estes AWS recursos oficiais:

- [Visualizando um manifesto de AWS Elemental MediaPackage- Use a visualização prévia do manifesto para solucionar problemas de empacotamento de conteúdo](#)
- [Aumente a taxa de acertos do CloudFront cache](#) - Guia abrangente para otimização do cache CDN
- [Filtragem de manifestos](#) - Guia completo da funcionalidade de MediaPackage filtragem

Integração AWS Elemental MediaTailor com a Amazon CloudFront

AWS Elemental MediaTailor se integra à Amazon CloudFront para melhorar o desempenho e a confiabilidade da entrega de conteúdo. CloudFront é uma rede de distribuição de conteúdo (CDN) que distribui seu conteúdo por meio de uma rede mundial de data centers chamada de pontos de presença. Quando os espectadores solicitam seu conteúdo MediaTailor, CloudFront encaminham as solicitações para o ponto de presença mais próximo. Essa abordagem reduz a latência e melhora o desempenho de seus espectadores.

A integração MediaTailor com CloudFront oferece vários benefícios importantes:

- Latência reduzida para os espectadores acessarem conteúdo personalizado
- Escalabilidade aprimorada para lidar com grandes públicos
- Confiabilidade aprimorada por meio de caminhos de entrega redundantes
- Otimização de custos por meio de estratégias eficientes de armazenamento em cache
- Recursos avançados, como failover multirregional com resiliência sensível à qualidade de mídia (MQAR)

Para obter informações abrangentes sobre os CloudFront recursos, consulte o [Guia do CloudFront desenvolvedor](#). Para obter informações sobre CloudFront preços, consulte [CloudFront Preços](#).

Para resumir, às vezes usamos “manifestos” para nos referirmos coletivamente a playlists multivariantes, playlists de mídia e. MPDs

Tópicos

- [Configure a CloudFront integração básica com MediaTailor](#)
- [Otimizando o MediaTailor desempenho com recursos CloudFront](#)
- [Implemente resiliência multirregional com o MQAR MediaTailor](#)
- [Monitore e solucione problemas de sua CloudFront integração MediaTailor](#)

Configure a CloudFront integração básica com MediaTailor

AWS Elemental MediaTailor a integração com CloudFront a Amazon melhora o desempenho da entrega de conteúdo para seus espectadores. Este tópico orienta você na configuração de uma CloudFront distribuição básica para MediaTailor. Com essa integração, seus espectadores podem

acessar conteúdo personalizado pela CloudFront rede. Você também aprenderá a configurar o armazenamento em cache adequado para diferentes tipos de conteúdo.

Para obter informações sobre como transmitir parâmetros de consulta CloudFront para autorização e roteamento, consulte [MediaTailor parâmetros de consulta do manifesto](#). Para roteamento avançado usando variáveis dinâmicas, consulte [MediaTailor variáveis de domínio para várias fontes de conteúdo](#).

Pré-requisitos

Antes de configurar CloudFront com MediaTailor, verifique se você tem o seguinte:

- Uma AWS conta ativa com permissões para criar e gerenciar CloudFront distribuições
- Uma configuração de MediaTailor reprodução configurada (consulte [Utilizar AWS Elemental MediaTailor para inserir anúncios](#))
- Seu servidor de origem de conteúdo devidamente configurado e acessível
- Compreensão básica dos conceitos de streaming de vídeo (HLS/DASH)

Configurando a distribuição CloudFront

Siga estas etapas para criar e configurar uma CloudFront distribuição para MediaTailor:

Para criar uma CloudFront distribuição para MediaTailor

1. Faça login no AWS Management Console e abra o CloudFront console em <https://console.aws.amazon.com/cloudfront/v3/home>.
2. Escolha Criar distribuição.
3. Para o domínio Origin, insira o URL do seu MediaTailor endpoint (por exemplo, `a1b2c3d4.mediatailor.us-west-2.amazonaws.com`).
4. Em Protocolo, selecione Somente HTTPS.
5. Em Nome, insira um nome que ajude a identificar essa origem (por exemplo, `mediatailor-origin`).
6. Defina as configurações padrão de comportamento do cache:
 - a. Para o padrão Path, use o valor padrão (*).
 - b. Em Comprimir objetos automaticamente, selecione Sim.

- c. Em Política de protocolo Viewer, selecione Redirecionar HTTP para HTTPS.
- d. Em Métodos HTTP permitidos, selecione GET, HEAD.
- e. Em Política de cache, selecione CachingDisabled.
- f. Para a política de solicitação do Origin, selecione AllViewerencaminhar todos os cabeçalhos para o comportamento padrão.

 Note

O comportamento padrão é usado AllViewer para lidar com segurança com qualquer conteúdo que não corresponda aos padrões de caminho específicos. Comportamentos de cache específicos para manifestos e segmentos serão configurados separadamente com as políticas apropriadas.

- 7. Defina as configurações de distribuição:
 - a. Para Classe de preço, selecione a opção que melhor corresponda às localizações do seu público.
 - b. Para AWS WAF Web ACL, selecione uma Web ACL existente ou deixe como Não habilitar proteções de segurança.
 - c. Para Objeto raiz padrão, deixe em branco.
 - d. Em Registro padrão, selecione Ativado para ativar o registro.
- 8. Escolha Criar distribuição.

Configurando comportamentos de cache

Depois de criar sua distribuição, você precisa configurar comportamentos adicionais de cache para lidar com diferentes tipos de conteúdo de forma adequada. Esta seção aborda a configuração básica do comportamento do cache para CloudFront.

Para uma otimização abrangente do cache, incluindo configurações avançadas de TTL, configurações de chave de cache e ajuste de desempenho, consulte [Otimização de cache para CDN e integrações MediaTailor](#) o guia de otimização da CDN.

Configurando o comportamento do cache do manifesto

Não armazene manifestos em cache porque eles contêm conteúdo personalizado. Siga estas etapas para configurar o comportamento do cache:

Para configurar o comportamento do cache do manifesto

1. No CloudFront console, selecione sua distribuição.
2. Escolha a guia Behaviors.
3. Escolha Create behavior (Criar comportamento).
4. Em Path pattern, insira `*.m3u8` para combinar as playlists multivariantes e de mídia do HLS.
5. Para Origin, selecione sua MediaTailor origem.
6. Em Política de cache, selecione CachingDisabled.
7. Para a política de solicitação do Origin, selecione AllViewerpara encaminhar todos os cabeçalhos necessários para conteúdo dinâmico.
8. Escolha Criar.
9. Repita essas etapas para manifestos do DASH usando o padrão de caminho correspondente `*.mpd`. MPDs

Essa configuração garante que cada espectador receba um manifesto personalizado com o conteúdo específico do anúncio. A CDN não armazena esses manifestos em cache, então cada solicitação vai diretamente para MediaTailor

Configurando o comportamento do cache de segmentos

Configure comportamentos de cache separados para segmentos de anúncios e segmentos de conteúdo para otimizar o desempenho e garantir o tratamento adequado do CORS.

Configurando o comportamento do cache do segmento de anúncios

Os segmentos de anúncios veiculados por meio do padrão de `/tm/*` caminho exigem uma configuração específica para lidar com o CORS adequadamente. Siga estas etapas:

Para configurar o comportamento do cache do segmento de anúncios

1. No CloudFront console, selecione sua distribuição.
2. Escolha a guia Behaviors.
3. Escolha Create behavior (Criar comportamento).
4. Em Padrão de caminho, insira `/tm/*` para corresponder aos segmentos de anúncios veiculados por MediaTailor.
5. Para Origin, selecione a origem MediaTailor dos seus segmentos (usando o `segments.mediatailor.region.amazonaws.com` nome do host).

6. Em Política de cache, selecione CachingOptimized.
7. Para Política de solicitação do Origin, selecione Nenhuma.
8. Para a política de cabeçalhos de resposta, selecione CORS- with-preflight-and - SecurityHeadersPolicy para garantir que os cabeçalhos CORS adequados sejam incluídos nas respostas.
9. Escolha Criar.

Configurando o comportamento do cache do segmento de conteúdo

Os segmentos de conteúdo podem usar políticas de armazenamento em cache padrão para um desempenho ideal. Configure comportamentos separados para diferentes formatos de segmento:

Para configurar o comportamento do cache do segmento de conteúdo

1. No CloudFront console, selecione sua distribuição.
2. Escolha a guia Behaviors.
3. Escolha Create behavior (Criar comportamento).
4. Em Padrão de caminho, insira `*.ts` para corresponder aos segmentos de conteúdo HLS.
5. Para Origin, selecione a origem do seu conteúdo.
6. Em Política de cache, selecione CachingOptimized.
7. Para Política de solicitação do Origin, selecione Nenhuma.
8. Para a política de cabeçalhos de resposta, selecione CORS- with-preflight-and - SecurityHeadersPolicy para garantir o tratamento consistente do CORS em todos os tipos de conteúdo.
9. Escolha Criar.
10. Repita essas etapas para outros formatos de segmento de conteúdo usando padrões de caminho apropriados:
 - `*.mp4` para MP4 segmentos
 - `*.m4s` para segmentos DASH
 - `*.cmfve` e `*.cmfa` para segmentos CMAF

Essa configuração garante que os segmentos de anúncios e segmentos de conteúdo sejam armazenados em cache adequadamente com o tratamento adequado do CORS. Os segmentos de

anúncios usam a origem dos MediaTailor segmentos com proteção CORS, enquanto os segmentos de conteúdo usam sua origem de conteúdo com políticas de armazenamento em cache otimizadas.

Atualizando a MediaTailor configuração

Depois de configurar sua CloudFront distribuição, atualize sua MediaTailor configuração para usar o CloudFront domínio:

Para atualizar sua MediaTailor configuração

1. Abra o [console do MediaTailor](#).
2. Selecione a configuração que você deseja atualizar.
3. Na seção Configuração da CDN, insira seu nome de domínio CloudFront de distribuição (por exemplo, `d1234abcdef.cloudfront.net`) no campo de prefixo do segmento de conteúdo da CDN.
4. Salve as alterações.

Com essa configuração, MediaTailor gera manifestos com URLs desse ponto para sua CloudFront distribuição em vez de diretamente para a origem.

Testando sua integração

Depois de configurar sua CloudFront distribuição e atualizar sua MediaTailor configuração, teste a integração:

Para testar sua CloudFront MediaTailor integração

1. Solicite um manifesto por meio CloudFront de sua distribuição (por exemplo, `https://d1234abcdef.cloudfront.net/v1/master/12345/my-config/index.m3u8`).
2. Verifique se o manifesto contém segmentos URLs que apontam para seu CloudFront domínio.
3. Reproduza o conteúdo por meio de um player de vídeo e verifique se o conteúdo e os anúncios são reproduzidos corretamente.
4. Verifique CloudFront os registros para garantir que as solicitações estejam sendo roteadas corretamente.

Exemplo de configuração

Aqui está um exemplo de uma configuração de CloudFront distribuição MediaTailor com comportamentos de cache adequados:

Example CloudFront exemplo de configuração de distribuição

```
{
  "DefaultCacheBehavior": {
    "TargetOriginId": "mediatailor-origin",
    "ViewerProtocolPolicy": "redirect-to-https",
    "AllowedMethods": {
      "Quantity": 2,
      "Items": ["GET", "HEAD"]
    },
    "CachePolicyId": "4135ea2d-6df8-44a3-9df3-4b5a84be39ad",
    "OriginRequestPolicyId": "59781a5b-3903-41f3-afcb-af62929ccde1",
    "Comment": "Default behavior with CachingDisabled and AllViewer"
  },
  "CacheBehaviors": [
    {
      "PathPattern": "/*.m3u8",
      "TargetOriginId": "mediatailor-origin",
      "ViewerProtocolPolicy": "redirect-to-https",
      "CachePolicyId": "4135ea2d-6df8-44a3-9df3-4b5a84be39ad",
      "OriginRequestPolicyId": "59781a5b-3903-41f3-afcb-af62929ccde1",
      "Comment": "Manifest behavior with CachingDisabled and AllViewer"
    },
    {
      "PathPattern": "/*.ts",
      "TargetOriginId": "mediatailor-origin",
      "ViewerProtocolPolicy": "redirect-to-https",
      "CachePolicyId": "658327ea-f89d-4fab-a63d-7e88639e58f6",
      "OriginRequestPolicyId": "88a5eaf4-2fd4-4709-b370-b4c650ea3fcf",
      "Comment": "Segment behavior with CachingOptimized and HostHeaderOnly"
    }
  ]
}
```

Esse exemplo mostra:

- Comportamento padrão: usa `CachingDisabled` e `AllViewer` para lidar com segurança com qualquer conteúdo que não corresponda a padrões de caminho específicos
- Comportamento manifesto (*.m3u8): usos e para conteúdo dinâmico `CachingDisabled` `AllViewer`
- Comportamento do segmento (*.ts): usos e `CachingOptimized` `CORS-with-preflight-and-SecurityHeadersPolicy`

Próximas etapas

Depois de configurar sua CloudFront integração básica com MediaTailor, considere estas próximas etapas:

- Otimize o desempenho com CloudFront recursos adicionais (consulte [Otimizando o MediaTailor desempenho com recursos CloudFront](#))
- Implemente resiliência multirregional com o MQAR (consulte) [Implemente resiliência multirregional com o MQAR MediaTailor](#)
- Configure o monitoramento e a solução de problemas (consulte [Monitore e solucione problemas de sua CloudFront integração MediaTailor](#))

Otimizando o MediaTailor desempenho com recursos CloudFront

AWS Elemental MediaTailor o desempenho com a Amazon CloudFront pode ser aprimorado por meio de recursos adicionais além da configuração básica. Depois de definir sua CloudFront configuração básica com MediaTailor, você pode implementar recursos adicionais para aprimorar o desempenho, a confiabilidade e as opções de personalização. Essas otimizações ajudam a oferecer uma melhor experiência de visualização. Eles também reduzem os custos e a carga de origem.

Para otimizar o desempenho usando roteamento dinâmico e tratamento de parâmetros, consulte [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#). Para estratégias de otimização de parâmetros de consulta, consulte [MediaTailor parâmetros de consulta do manifesto](#).

Reduzindo a carga de origem com o Origin Shield

O Origin Shield adiciona uma camada de cache entre os pontos CloudFront de presença e seu servidor de origem. Esse recurso é valioso para transmissão ao vivo e conteúdo VOD popular. Isso ajuda quando muitos espectadores solicitam o mesmo conteúdo ao mesmo tempo.

Ao consolidar solicitações de vários pontos de presença, o Origin Shield reduz a carga MediaTailor e as origens do seu conteúdo.

Para habilitar o Origin Shield para sua MediaTailor origem

1. Abra o CloudFront console e navegue até sua distribuição.
2. Selecione a origem que aponta para sua configuração de MediaTailor reprodução.
3. Em Origin Shield, selecione Sim.
4. No menu suspenso, selecione a AWS região mais próxima do seu MediaTailor endpoint.
5. Salve as alterações.

Para eventos de alto tráfego, o Origin Shield reduz significativamente a carga de solicitações em sua origem e melhora a confiabilidade. Para obter instruções detalhadas, consulte Como [ativar o Origin Shield](#) no guia do CloudFront desenvolvedor.

Personalizando a entrega de conteúdo com CloudFront o Functions

CloudFront As funções permitem que você execute um JavaScript código leve na borda para modificar as solicitações e respostas do visualizador. Você pode usar essas funções para personalizações simples, como modificações de URL, manipulação de cabeçalhos ou autenticação básica. Para MediaTailor fluxos de trabalho, as Funções ajudam com tarefas que não exigem processamento complexo.

CloudFront As funções fornecem uma maneira leve de personalizar a entrega de conteúdo na borda. Veja como implementá-los para sua MediaTailor integração:

Para implementar CloudFront funções para MediaTailor

1. No CloudFront console, navegue até Funções.
2. Crie uma nova função e selecione a finalidade apropriada:
 - Manipulação de URL - Para modificar a lista de reprodução multivariada, a lista de reprodução de mídia e a solicitação MPD antes que elas cheguem URLs MediaTailor
 - Manipulação de cabeçalho - Para adicionar ou modificar cabeçalhos de solicitação
 - Autenticação simples - Para validar tokens ou parâmetros de consulta
3. Escreva o código JavaScript da função.
4. Teste sua função com MediaTailor solicitações de amostra.

5. Publique e associe a função ao comportamento de cache da sua distribuição.

Example CloudFront Função de amostra para normalização de URL

```
function handler(event) {
    var request = event.request;
    var uri = request.uri;

    // Normalize URLs to lowercase to improve cache hit ratio
    if (uri.includes('.m3u8') || uri.includes('.mpd')) {
        request.uri = uri.toLowerCase();
    }

    return request;
}
```

Para obter mais informações e exemplos de código, consulte [Personalizar na borda usando CloudFront funções](#) no guia do CloudFront desenvolvedor.

Implementando personalizações avançadas com o Lambda @Edge

Quando você precisar de recursos de processamento mais complexos do que o CloudFront Functions pode fornecer, use o Lambda @Edge. Esse serviço permite que você execute funções do Node.js ou do Python em locais CloudFront periféricos.

As funções do Lambda @Edge podem realizar operações sofisticadas, como autenticação complexa, modificações maiores na resposta ou integrações de API de terceiros.

Para personalizações mais complexas, use as funções do Lambda @Edge com sua integração e: MediaTailor CloudFront

Para implementar o Lambda @Edge com MediaTailor

1. Crie uma função do Lambda na região Leste dos EUA (N. da Virgínia).
2. Escreva seu código de função para um desses casos de uso:
 - Manipulação de URL - Para modificar a lista de reprodução multivariada, a lista de reprodução de mídia e a solicitação MPD antes que elas cheguem URLs MediaTailor

- Teste A/B - Para direcionar os usuários para diferentes servidores de decisão de anúncios
 - Solicitar autenticação - Para adicionar cabeçalhos de autenticação
 - Modificação do cabeçalho de resposta - Para adicionar cabeçalhos CORS
3. Publique uma versão da sua função e crie um alias de função.
 4. Associe a função à sua CloudFront distribuição no ponto de gatilho apropriado (solicitação do espectador ou resposta do espectador).

Note

Ao usar o Lambda @Edge com MediaTailor, evite usar acionadores voltados para a origem (solicitação de origem e resposta de origem) se você planeja usar os recursos de resiliência sensível à qualidade de mídia (MQAR), pois eles são incompatíveis no momento.

Para obter mais informações e exemplos de código, consulte [Personalizar na borda usando o Lambda @Edge no guia](#) do CloudFront desenvolvedor.

Dicas adicionais de otimização de desempenho

Considere essas otimizações adicionais para melhorar ainda mais o desempenho:

Otimize as taxas de acerto do cache

Monitore sua taxa de acertos de cache nas CloudFront métricas e procure oportunidades para melhorá-la:

- Padronize os padrões de URL para melhorar a consistência da chave de cache
- Use a lista branca de cadeias de caracteres de consulta para incluir somente os parâmetros necessários na chave de cache
- Considere implementar a normalização de URL com Funções CloudFront

Reduza a latência

Implemente essas técnicas para minimizar a latência:

- Ative a compressão Brotli para respostas baseadas em texto
- Use HTTP/2 ou HTTP/3 para melhorar a eficiência da conexão
- Considere habilitar IPv6 o suporte para redes modernas

Otimização de custos

Equilibre o desempenho com a eficiência de custos:

- Use o Origin Shield para reduzir solicitações de origem redundantes
- Considere a seleção da classe de preço com base na geografia do seu público
- Implemente armazenamento em cache agressivo para segmentos para reduzir o tráfego de origem

Próximas etapas

Depois de otimizar o desempenho com CloudFront recursos, considere estas próximas etapas:

- Implemente resiliência multirregional com o MQAR (consulte [Implemente resiliência multirregional com o MQAR MediaTailor](#))
- Configure o monitoramento e a solução de problemas (consulte [Monitore e solucione problemas de sua CloudFront integração MediaTailor](#))

Implemente resiliência multirregional com o MQAR MediaTailor

AWS Elemental MediaTailor A resiliência multirregional é aprimorada por meio do Media Quality-Aware Resiliency (MQAR), um CloudFront recurso avançado da Amazon que ajuda a MediaTailor oferecer a melhor experiência de streaming possível. Ele seleciona automaticamente a origem com o maior índice de qualidade quando você tem várias origens em diferentes AWS regiões. Esse recurso é particularmente valioso para transmissão ao vivo, onde você precisa de um serviço ininterrupto.

Como funciona o MQAR

O CloudFront MQAR permite selecionar automaticamente a origem com o maior índice de qualidade. Isso garante que os espectadores recebam a melhor experiência de streaming possível. Quando configurado corretamente, o MQAR oferece os seguintes benefícios:

- Seleção automática da origem da mais alta qualidade
- Failover contínuo entre AWS regiões durante interrupções
- Experiência aprimorada do espectador com o mínimo de interrupções
- Monitoramento e adaptação da qualidade em tempo real

Etapa 1: Verificar os requisitos do MQAR

Antes de implementar o MQAR, verifique se sua infraestrutura atende a esses requisitos. O MQAR funciona comparando índices de qualidade de várias origens, então você precisa de conteúdo idêntico disponível em várias AWS regiões.

- Codificadores que enviam fluxos de ingestão alinhados para todos os MediaPackage canais usando conjuntos de ingestão CMAF bloqueados por época
- Dois MediaPackage canais idênticos em AWS regiões diferentes, com endpoints de origem idênticos
- Ingestão de CMAF para MediaPackage canais (ativada por padrão)
- CloudFront distribuição configurada para suportar MQAR
- MediaTailor configurações para cada endpoint MediaPackage

Etapa 2: Configurar seus codificadores para MQAR

Seus codificadores precisam produzir saídas consistentes e sincronizadas em todas as regiões para que o MQAR funcione de forma eficaz. Essa consistência permite CloudFront fazer comparações precisas de qualidade entre as origens.

Configure suas MediaLive saídas com estas configurações:

- Certifique-se de que todas as taxas de quadros de vídeo no grupo de saída CMAF sejam consistentes (todas as taxas de quadros fracionárias ou todas as taxas de quadros inteiras).
- Evite transições entre taxas de quadros fracionárias e inteiras nas sessões de codificação.
- Configure os números de sequência do segmento de saída para que eles nunca retrocedam nas sessões de codificação.
- Use nomes de saída de codificador idênticos em todas as regiões.

Para obter mais informações sobre a configuração MediaLive do MQAR, consulte [Trabalhando com o MQCS no guia](#) do MediaLive usuário.

Etapa 3: Configurar MediaPackage para MQAR

Configure seus MediaPackage canais e endpoints com estas configurações:

Para configurar MediaPackage para MQAR

1. Crie configurações idênticas de canais e endpoints em cada AWS região.
2. Use CMAF como o tipo de entrada do canal.
3. Para a MediaPackage origem primária, habilite Forçar configuração de erro do endpoint com estas configurações:
 - Playlists multivariantes obsoletas, playlists de mídia ou MPDs
 - Lista de reprodução multivariante incompleta, lista de reprodução de mídia ou MPD
 - Entrada de ardósia
4. Para MediaPackage origens de backup, não ative essas configurações de erro para maximizar as chances de um failover bem-sucedido.

Para obter mais informações sobre a configuração MediaPackage do MQAR, consulte [Aproveitando os índices de qualidade de mídia usando AWS Elemental MediaPackage no guia](#) do MediaPackage usuário.

Etapa 4: Configurar CloudFront para MQAR

Na CloudFront configuração, você ativa o MQAR e define como ele seleciona entre suas origens. Crie um grupo de origem com a opção Pontuação de qualidade de mídia ativada.

Para configurar CloudFront para MQAR

1. Crie uma CloudFront distribuição com origens apontando para seus MediaTailor endpoints.
2. Crie um grupo de origem que inclua essas origens.
3. Nas configurações do grupo de origem, ative a opção Pontuação de qualidade de mídia.
4. Configure os critérios de failover para incluir códigos de resposta 404 Not Found. Opcionalmente, você pode incluir outros códigos 4xx/5xx.
5. Crie comportamentos de cache separados para o padrão de caminho de cada canal. Isso evita a mistura de partituras quando o mesmo grupo de origem atende a vários canais.

 Note

O MQAR não está disponível ao usar as funções do Lambda @Edge em acionadores voltados para a origem (solicitação de origem e resposta de origem) associados ao comportamento de cache da sua distribuição.

Para obter mais informações sobre a configuração do CloudFront MQAR, consulte [Resiliência com reconhecimento de qualidade de mídia](#) no guia do desenvolvedor. CloudFront

Etapa 5: Configurar MediaTailor para MQAR

Para concluir a configuração do MQAR, configure MediaTailor em cada região para trabalhar com sua arquitetura multirregional. Isso garante a inserção consistente do anúncio, independentemente da origem CloudFront selecionada.

Para configurar MediaTailor para MQAR

1. Crie MediaTailor configurações idênticas em cada região, apontando para os MediaPackage endpoints correspondentes.
2. Configure o prefixo do segmento de conteúdo CDN para usar seu domínio de CloudFront distribuição.
3. Certifique-se de que as configurações do servidor de decisão de anúncios sejam idênticas em todas as regiões.

Essa configuração garante que, independentemente da origem CloudFront selecionada com base nos índices de qualidade, MediaTailor possa continuar personalizando anúncios de forma consistente.

Etapa 6: Teste sua configuração de MQAR

Depois de configurar o MQAR, teste sua configuração para garantir que ela funcione conforme o esperado:

Para testar sua configuração MQAR

1. Solicite conteúdo por meio de sua CloudFront distribuição.
2. Monitore os registros em tempo real para verificar se CloudFront está selecionando as origens com base nos índices de qualidade.

3. Simule uma falha na sua AWS região principal para testar o comportamento do failover.
4. Verifique se a inserção de anúncios continua funcionando corretamente durante o failover.

Próximas etapas

Depois de implementar o MQAR, considere estas próximas etapas:

- Configure o monitoramento e a solução de problemas para sua configuração MQAR (consulte [Monitore e solucione problemas de sua CloudFront integração MediaTailor](#))
- Implemente a implantação automatizada usando AWS CloudFormation (consulte [Automatize MediaTailor e CDN com CloudFormation](#))

Monitore e solucione problemas de sua CloudFront integração MediaTailor

AWS Elemental MediaTailor a integração com a Amazon CloudFront exige monitoramento e solução de problemas contínuos para manter o desempenho ideal. Depois de implementar sua MediaTailor integração CloudFront e integração, monitore o desempenho e prepare-se para solucionar qualquer problema. A rede de distribuição de conteúdo (CDN) fornece ferramentas para ajudá-lo a entender o desempenho de sua distribuição e identificar possíveis problemas.

Configurando o monitoramento para sua integração

Implemente essas estratégias de monitoramento para monitorar o desempenho CloudFront de sua MediaTailor integração:

Para configurar o monitoramento para sua CloudFront MediaTailor integração

1. Ativar o registro CloudFront padrão:
 - a. No CloudFront console, selecione sua distribuição.
 - b. Escolha a guia Logs.
 - c. Em Registros padrão, escolha Editar.
 - d. Selecione Ativado e configure um bucket do Amazon S3 para armazenamento de registros.
 - e. Escolha Salvar alterações.
2. Configure registros CloudFront em tempo real:
 - a. No CloudFront console, selecione sua distribuição.

- b. Escolha a guia Logs.
 - c. Em Registros em tempo real, escolha Editar.
 - d. Selecione Ativado e configure um stream de entrega do Amazon Kinesis Data Streams ou do Amazon Data Firehose.
 - e. Inclua esses campos em sua configuração de registro:
 - `time-to-first-byte`- Tempo de resposta
 - `sc-status`- Código de status HTTP
 - `c-ip`- Endereço IP do visualizador
 - `cs-uri-stem`- Solicitar caminho do URI
 - `cs-user-agent`- Agente de usuário
 - `x-edge-result-type`- Tipo de resultado (acerto, erro, erro)
 - f. Escolha Salvar alterações.
3. Crie CloudWatch painéis:
 - a. No CloudWatch console, escolha Painéis.
 - b. Escolha Create dashboard (Criar painel).
 - c. Adicione widgets para essas CloudFront métricas:
 - Solicitações
 - BytesDownloaded
 - 4 xxErrorRate
 - 5xxErrorRate
 - TotalErrorRate
 - CacheHitRate
4. Configurar CloudWatch alarmes:
 - a. No CloudWatch console, escolha Alarmes.
 - b. Selecione Criar alarme.
 - c. Crie alarmes para estas condições:
 - 5 xxErrorRate > 1% por 5 minutos
 - 4 xxErrorRate maior que 5% por 5 minutos

- CacheHitRate menos de 80% por 30 minutos

Monitorando o desempenho do MQAR

Ao usar o MQAR, o monitoramento ajuda você a entender como CloudFront seleciona entre suas origens e se os índices de qualidade atendem às expectativas. Registros em tempo real mostram essas decisões à medida que elas acontecem.

Para monitorar o desempenho do MQAR

1. Habilite registros em tempo real para sua CloudFront distribuição.
2. Inclua esses campos em sua configuração de registro:
 - `r-host`- O nome do host da origem selecionada
 - `sr-reason`- O motivo da seleção da origem
 - `x-edge-mqcs`- A pontuação de confiança na qualidade da mídia
3. Configure um destino de log no Amazon Kinesis Data Streams ou no Amazon Data Firehose.
4. Crie painéis ou alertas com base nessas métricas para monitorar os índices de qualidade e as decisões de seleção de origem.

Example CloudWatch Painel de amostra para monitoramento de MQAR

Crie um CloudWatch painel com essas métricas:

- A seleção de origem conta por região
- Índices de qualidade médios ao longo do tempo
- Eventos de failover
- Taxas de erro de 4xx e 5xx por origem

Para obter mais informações sobre como configurar registros em tempo real, consulte [Registros em tempo real](#) no guia do CloudFront desenvolvedor.

Solução de problemas comuns do

Podem surgir problemas com sua CloudFront MediaTailor integração, mesmo com um planejamento cuidadoso. Compreender problemas e soluções comuns ajuda você a resolver problemas rapidamente e minimizar o impacto do espectador.

Se você encontrar problemas com sua MediaTailor integração CloudFront e integração, verifique estes problemas e soluções comuns:

Problemas de cache manifestos

Sintoma: manifestos obsoletos ou anúncios não atualizados

Solução: verifique se você está usando a política de `CachingDisabled` cache para playlists multivariantes, playlists de mídia e caminhos MPD. Verifique se sua configuração encaminha os parâmetros de consulta corretamente.

Erros CORS

Sintoma: o console do navegador mostra erros de CORS ao acessar o conteúdo

Solução: configure uma política de cabeçalhos de resposta com cabeçalhos CORS apropriados e associe-a ao seu comportamento de cache.

MQAR não está funcionando

Sintoma: a seleção de origem ignora os índices de qualidade

Solução: verifique se você ativou a opção Índice de qualidade de mídia nas configurações do seu grupo de origem. Verifique se você não está usando gatilhos voltados para a origem do Lambda @Edge.

Falhas na inserção de anúncios

Sintoma: os anúncios não estão sendo inseridos corretamente

Solução: verifique seus pontos MediaTailor de configuração para sua CloudFront distribuição quanto ao prefixo do segmento de conteúdo. Verifique se sua configuração encaminha todos os cabeçalhos necessários.

Para problemas mais complexos, você pode usar estas abordagens de solução de problemas:

1. Verifique os registros CloudFront de distribuição quanto a padrões de erro
2. Use as ferramentas do desenvolvedor do navegador para inspecionar solicitações de rede
3. Compare o conteúdo do manifesto MediaTailor diretamente com o conteúdo entregue por meio de CloudFront
4. Teste com um player simples que suporta registro detalhado

Para obter mais assistência na solução de problemas, consulte a seção [Solução](#) de problemas no guia MediaTailor do usuário.

Solução de problemas de trabalho

Siga esta abordagem sistemática para solucionar problemas com sua CloudFront MediaTailor integração:

Para solucionar problemas CloudFront e problemas de MediaTailor integração

1. Identifique o problema específico:
 - a. Determine se o problema afeta todos os espectadores ou apenas alguns
 - b. Identifique quais tipos de conteúdo são afetados (manifestos, segmentos ou ambos)
 - c. Anote quaisquer mensagens de erro ou sintomas
2. CloudFront Registros de verificação:
 - a. Procure códigos de status de erro (4xx ou 5xx)
 - b. Verifique os hit/miss padrões de cache
 - c. Verifique se as solicitações estão sendo encaminhadas para a origem correta
3. Verifique a configuração:
 - a. Verifique os comportamentos do cache para obter os padrões de caminho corretos
 - b. Verifique se as políticas de cache estão aplicadas corretamente
 - c. Confirme se as políticas de solicitação de origem estão encaminhando os cabeçalhos necessários
4. Teste o acesso direto:
 - a. Tente acessar o conteúdo diretamente de MediaTailor (ignorando CloudFront)
 - b. Compare as respostas entre acesso direto e CloudFront acesso
5. Implemente a solução:
 - a. Aplique a correção apropriada com base em suas descobertas
 - b. Teste para verificar se o problema foi resolvido
 - c. Documente o problema e a solução para referência futura

Próximas etapas

Depois de configurar o monitoramento e a solução de problemas para sua CloudFront MediaTailor integração, considere estas próximas etapas:

- Implemente a implantação automatizada usando AWS CloudFormation (consulte [Automatize MediaTailor e CDN com CloudFormation](#))
- Crie runbooks para cenários operacionais comuns e procedimentos de solução de problemas
- Configure a remediação automatizada para problemas comuns

Configure terceiros CDNs para entrega MediaTailor de anúncios

Terceiros CDNs como a Akamai e a Fastly podem melhorar significativamente o desempenho e a escalabilidade da entrega de AWS Elemental MediaTailor anúncios e, ao mesmo tempo, reduzir os custos de largura de banda. No entanto, a configuração de CDN para publicidade personalizada requer configurações específicas que diferem da entrega de vídeo padrão.

Para obter informações sobre como transmitir parâmetros de consulta por meio de terceiros CDNs, consulte [MediaTailor parâmetros de consulta do manifesto](#). Para configurações avançadas de roteamento usando variáveis dinâmicas, consulte. [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#)

Este guia explica o processo completo de configuração de sua CDN de terceiros para que ela funcione de forma otimizada. MediaTailor Você aprenderá a configurar dois comportamentos essenciais:

- Ignorar manifesto: garante que cada espectador receba inserções de anúncios personalizadas, evitando o armazenamento em cache do manifesto
- Cache de segmentos: maximiza o desempenho e reduz os custos ao armazenar em cache o conteúdo de vídeo de forma eficiente

O processo de configuração normalmente leva de 30 a 60 minutos e inclui seleção, configuração, verificação e otimização do provedor. Depois de concluída, você terá uma configuração de CDN que oferece anúncios personalizados de forma eficiente, mantendo a experiência ideal do espectador.

 Note

Este guia se concentra nas configurações da Akamai e do Fastly. Para obter instruções de CloudFront configuração, consulte [Integração do CloudFront](#).

Pré-requisitos

Antes de configurar sua CDN de terceiros com MediaTailor, verifique se você tem:

- Uma MediaTailor configuração ativa que inclui a origem do conteúdo e o servidor de decisão do anúncio
- Acesso à interface de configuração da sua CDN
- Uma lista de extensões de arquivo usadas em seu conteúdo (.m3u8, .mpd, .ts e assim por diante)
- Documentação do seu provedor de CDN para referência

Para obter instruções CloudFront de configuração em vez de instruções de terceiros CDNs, consulte [Integração do CloudFront](#).

Terminologia

Para entender os requisitos de configuração da CDN, você precisa conhecer esses tipos de manifesto:

- O HLS manifesta:
 - Playlist multivariante: o manifesto de nível superior que contém links para playlists de mídia
 - Playlist de mídia: o manifesto de segundo nível com links para segmentos de conteúdo
- O DASH manifesta:
 - MPD (Descrição da apresentação de mídia): o termo padrão para manifestos do DASH

Este guia se refere a todos os arquivos de manifesto (playlists multivariantes, playlists de mídia e MPDs) coletivamente como manifestos ao discutir requisitos comuns de configuração.

Para os princípios gerais de configuração de CDN que se aplicam a todos os provedores, consulte [Configure a integração da CDN com MediaTailor](#).

Para obter orientação sobre otimização de CDN, consulte [Guia de otimização de desempenho para CDN e integrações MediaTailor](#).

Configurar regras de armazenamento em cache da CDN

A configuração de cache da CDN é fundamental para a entrega de MediaTailor anúncios, pois determina como seu conteúdo alcança os espectadores. A configuração adequada garante que os manifestos permaneçam personalizados para cada visualizador, enquanto os segmentos são armazenados em cache de forma eficiente para reduzir a carga de origem e melhorar o desempenho.

Essa configuração normalmente leva de 15 a 30 minutos por provedor de CDN e requer dois comportamentos distintos:

- Manipulação de manifestos: impede o armazenamento em cache para garantir que cada espectador receba inserções de anúncios personalizadas
- Cache de segmentos: maximiza a eficiência do cache de conteúdo de vídeo para melhorar o desempenho de entrega

Siga estas etapas para configurar as regras de cache da sua CDN para otimizar a entrega de anúncios.

Escolha seu provedor de CDN nas guias a seguir para obter instruções específicas:

Akamai

Configure esses dois comportamentos em sua propriedade da Akamai:

- Manipulação de manifestos para evitar o armazenamento em cache
- Cache de segmentos para desempenho ideal

Configurar a entrega do manifesto

Configure sua CDN da Akamai para evitar o armazenamento em cache de manifestos para que cada espectador receba anúncios personalizados.

Os arquivos de manifesto contêm os pontos de inserção de anúncios personalizados que são MediaTailor gerados para cada espectador. O armazenamento em cache desses arquivos faria

com que todos os espectadores vissem anúncios idênticos, anulando o propósito da publicidade personalizada.

Siga estas etapas para solicitações de manifesto (arquivos que terminam em .m3u8, .mpd ou .smil):

1. Crie um comportamento que corresponda às extensões do arquivo de manifesto (.m3u8, .mpd, .smil)
2. Defina a opção de armazenamento em cache como Sem armazenamento
3. Configure as chaves de cache para incluir todos os parâmetros de consulta
4. Ative o cabeçalho de encaminhamento do host para o roteamento de origem adequado
5. Configure o encaminhamento de cabeçalhos para todos os cabeçalhos. Para obter os requisitos mínimos, consulte [Cabeçalhos necessários para integração com MediaTailor CDN](#).

Configurar a entrega por segmento

Configure sua CDN da Akamai para armazenar segmentos de vídeo em cache para maximizar a eficiência da CDN e reduzir a carga de origem.

Os segmentos de vídeo são os arquivos de conteúdo reais que podem ser armazenados em cache com segurança porque não contêm informações personalizadas. O cache adequado de segmentos reduz os custos de largura de banda e melhora o desempenho da reprodução para os espectadores.

Siga estas etapas para solicitações de segmentos (arquivos que terminam em .ts, .mp4, .m4s etc.):

1. Crie um comportamento que corresponda às extensões de arquivo do segmento (.ts, .mp4, .m4s)
2. Defina o Honor Origin Cache Control como Sim
3. Defina as configurações padrão time-to-live (TTL) para quando os cabeçalhos de origem estiverem ausentes:
 - TTL padrão: 86400 segundos (24 horas)
 - TTL máximo: 604800 segundos (7 dias)

 Note

Depois de configurar esses comportamentos, ative suas alterações de propriedade no Akamai Control Center.

As alterações entram em vigor após a ativação.

Fastly

Crie essas duas configurações em seu serviço Fastly:

- Manipulação de manifestos para evitar o armazenamento em cache
- Cache de segmentos para desempenho ideal

Configurar a entrega do manifesto

Configure seu Fastly CDN para ignorar o armazenamento em cache de arquivos de manifesto para que cada espectador receba conteúdo de anúncio personalizado.

Os arquivos de manifesto devem estar MediaTailor disponíveis para cada solicitação para garantir a personalização adequada do anúncio. Ignorar o cache desses arquivos garante que os parâmetros de segmentação exclusivos de cada visualizador sejam processados corretamente.

Siga estas etapas para solicitações de manifesto:

1. Crie uma condição de solicitação para identificar caminhos de manifesto
2. Defina a condição de cache como Não armazenar em cache para essas solicitações
3. Defina as configurações de encaminhamento para incluir todos os parâmetros de consulta
4. Adicione User-Agent à sua configuração de encaminhamento de cabeçalho

Configurar a entrega por segmento

Configure seu Fastly CDN para armazenar segmentos de vídeo em cache para melhorar o desempenho de entrega e reduzir o tráfego de origem.

O armazenamento em cache de segmentos é essencial para uma entrega econômica e uma experiência ideal para o espectador. Esses arquivos são idênticos para todos os visualizadores e se beneficiam significativamente do cache do CDN.

Siga estas etapas para solicitações de segmentos:

1. Crie uma condição de solicitação para identificar caminhos de segmentos
2. Defina as configurações de cache para honrar os cabeçalhos de cache de origem
3. Configure o padrão time-to-live (TTL) para 86400 segundos (24 horas) para quando os cabeçalhos de origem estiverem ausentes

 Note

Depois de fazer essas alterações, ative uma nova versão do seu serviço Fastly. A configuração entra em vigor após a ativação.

Verifique sua configuração de CDN

A verificação garante que sua configuração de CDN funcione corretamente antes de você direcionar o tráfego de produção por meio dela. Esses testes confirmam que a personalização de anúncios funciona adequadamente e que o armazenamento em cache fornece os benefícios de desempenho esperados.

A verificação completa normalmente leva de 10 a 15 minutos e deve ser realizada em várias localizações geográficas, se possível.

Depois de configurar sua CDN, execute essas verificações para verificar se ela está funcionando corretamente:

1. Teste a personalização do manifesto:
 - a. Solicite o mesmo URL de conteúdo com parâmetros de anúncio diferentes
 - b. Verifique se cada solicitação retorna inserções de anúncios diferentes
2. Teste o cache do segmento:
 - a. Verifique as métricas da CDN para ver a taxa de acertos do cache do segmento (deve ser maior que 90%)
 - b. Monitore o tráfego de origem para confirmar que é menor do que a entrega direta
3. Teste o desempenho da reprodução:
 - a. Reproduza conteúdo por meio de sua CDN de diferentes locais
 - b. Verifique a reprodução suave sem problemas de buffer

Para metodologias de teste abrangentes e procedimentos avançados de validação, consulte [Teste e validação para CDN e integrações MediaTailor](#).

Otimize o desempenho da CDN

Depois de verificar sua configuração básica, implemente essas otimizações para maximizar o desempenho e minimizar os custos:

Monitore as principais métricas de desempenho

Acompanhe essas métricas para garantir o desempenho ideal:

Taxa de acertos do cache

Meta: Mais de 90% para segmentos de vídeo

Impacto: taxas mais altas reduzem a carga de origem e melhoram a experiência do espectador

Monitore: verifique o painel de análise do seu provedor de CDN diariamente

Tempo de resposta do Origin

Meta: menos de 200 ms para solicitações de manifesto

Impacto: a entrega mais rápida do manifesto reduz o tempo de inicialização dos espectadores

Monitor: configure alertas para tempos de resposta superiores a 500 ms

Taxas de erro

Meta: Menos de 0,1% para todas as solicitações

Impacto: altas taxas de erro indicam problemas de configuração ou problemas de origem

Monitor: configure alertas para taxas de erro superiores a 1%

Ajuste o comportamento de armazenamento em cache

Ajuste essas configurações com base nas características do seu conteúdo e nos padrões de visualização:

Otimização do TTL do segmento

Conteúdo ao vivo: use TTL mais curto (1-4 horas) para garantir atualizações oportunas

Conteúdo VOD: use TTL mais longo (24 a 48 horas) para maximizar a eficiência do cache

Segmentos de anúncios: considere um TTL mais curto (30 minutos a 2 horas) para conteúdo de anúncio atualizado com frequência

Para recomendações abrangentes de TTL e estratégias de armazenamento em cache em todos os MediaTailor fluxos de trabalho, consulte. [Otimização de cache para CDN e integrações MediaTailor](#)

Otimização geográfica

Origens multirregionais: configure a seleção de origem com base na localização do visualizador

Localizações periféricas: habilite localizações de borda adicionais em regiões com alta concentração de espectadores

Failover: configure as origens de backup para alta disponibilidade

Otimize custos

Implemente essas estratégias para reduzir os custos da CDN e, ao mesmo tempo, manter o desempenho:

- Compressão: habilite a compactação gzip para arquivos de manifesto para reduzir o uso da largura de banda
- Estratégia de limpeza: implemente a limpeza seletiva do cache em vez da limpeza completa do cache
- Análise de tráfego: analise os padrões de tráfego mensalmente para identificar oportunidades de otimização
- Seleção de níveis: use níveis de serviço de CDN apropriados com base em seus requisitos de desempenho

Solucionar problemas de CDN de terceiros

Os problemas de configuração da CDN geralmente se manifestam como problemas de personalização de anúncios ou degradação do desempenho. Use essa abordagem sistemática para identificar e resolver os problemas mais comuns que afetam a entrega de MediaTailor anúncios.

A maioria das soluções de problemas pode ser concluída em 15 a 30 minutos, verificando os sintomas específicos e aplicando as soluções correspondentes.

Se os espectadores tiverem problemas com a entrega do anúncio ou com a qualidade da reprodução, use este guia para identificar e resolver problemas comuns de configuração de CDN:

Akamai: manifestos em cache

Sintoma: os espectadores veem anúncios idênticos mesmo quando você configura parâmetros de segmentação diferentes.

Solução: verifique se você aplicou a opção Sem armazenamento em cache aos caminhos do manifesto.

Verifique também se você incluiu parâmetros de consulta na chave de cache.

Fastly: chaves de cache incorretas

Sintoma: os espectadores experimentam uma personalização de anúncios inconsistente.

Os espectadores também podem ver anúncios destinados a outros espectadores.

Solução: verifique se você definiu as configurações de encaminhamento para incluir todos os parâmetros de consulta na chave de cache.

Geral: Alto tráfego de origem

Sintoma: seus servidores de origem experimentam um tráfego inesperadamente alto

Solução: verifique as configurações de cache do segmento e os valores time-to-live (TTL).

Verifique as taxas de acerto do cache em suas métricas de CDN.

Geral: erros de reprodução

Sintoma: os espectadores enfrentam falhas no buffer ou na reprodução

Solução: verifique as regras de roteamento da CDN e a integridade da origem.

Verifique se todos os cabeçalhos necessários estão sendo encaminhados corretamente.

Note

Se essas soluções não resolverem seu problema, verifique a documentação do seu provedor de CDN. Você também pode entrar em contato com a equipe de suporte para obter etapas adicionais de solução de problemas.

Para obter orientações gerais sobre solução de problemas de CDN, consulte [Solucionar problemas com a integração com MediaTailor a CDN](#).

Guia de otimização de desempenho para CDN e integrações MediaTailor

AWS Elemental MediaTailor o desempenho pode ser maximizado por meio da otimização sistemática da rede de entrega de conteúdo (CDN). Se você está implementando inserção de anúncios no lado do servidor (SSAI), montagem de canais ou fluxos de trabalho combinados, os princípios de otimização e as metas de desempenho permanecem consistentes. Este guia fornece técnicas de otimização e benchmarks abrangentes que se aplicam a todas as MediaTailor implementações.

Para otimização avançada de roteamento usando variáveis dinâmicas e aliases de configuração, consulte [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#). Para estratégias de otimização de parâmetros de consulta, consulte [MediaTailor parâmetros de consulta do manifesto](#).

Visão geral do fluxo de trabalho de otimização:

1. Configurar o armazenamento em cache - Defina valores de TTL e comportamentos de cache apropriados
2. Otimize o roteamento - configure o roteamento de solicitações e as políticas de origem
3. Avalie o desempenho - acompanhe os benchmarks estabelecidos
4. Aplique técnicas avançadas - Implemente recursos adicionais de otimização

Tópicos

- [Otimização de cache para CDN e integrações MediaTailor](#)
- [Otimização de roteamento de solicitações para CDN e integrações MediaTailor](#)
- [Benchmarks de desempenho para CDN e integrações MediaTailor](#)
- [Técnicas avançadas de otimização para CDN e integrações MediaTailor](#)

Otimização de cache para CDN e integrações MediaTailor

AWS Elemental MediaTailor os requisitos de armazenamento em cache variam de acordo com o tipo de fluxo de trabalho e o formato do conteúdo. A configuração adequada do armazenamento em

cache é essencial para o desempenho ideal, a eficiência de custos e a experiência do espectador. As seções a seguir fornecem orientações detalhadas de armazenamento em cache para diferentes MediaTailor implementações.

Cache de inserção de anúncios do lado do servidor (SSAI)

Para fluxos de trabalho de inserção de anúncios do lado do servidor, manifestos personalizados não podem ser armazenados em cache, mas o conteúdo e os segmentos de anúncios devem ser armazenados em cache de forma agressiva:

Configurações de cache SSAI CDN

Tipo de conteúdo	TTL	Padrão de caminho	Elementos principais do cache
Playlists multivariantes	0 segundos	/v1/master/ *	Caminho do URL + todos os parâmetros de consulta
Playlists de mídia	0 segundos	/v1/manifest/ *	Caminho do URL + todos os parâmetros de consulta
TRAVESSIA MPDs	0 segundos	/v1/dash/ *	Caminho do URL + todos os parâmetros de consulta
Segmentos de conteúdo	Mais de 24 horas	Caminhos específicos de conteúdo	Somente caminho de URL
Segmentos de anúncios	Mais de 24 horas	/v1/segmento/ *	Somente caminho de URL

- Defina TTL de 10 segundos para manifestos personalizados para garantir que os espectadores up-to-date recebam o conteúdo do anúncio
- Configure valores de TTL mais longos para segmentos de conteúdo e anúncios para maximizar a eficiência do cache
- Configure comportamentos de cache que incluam parâmetros de personalização na chave de cache se você oferecer suporte à publicidade direcionada

- Implemente o colapso de solicitações no nível da CDN para lidar com solicitações simultâneas com eficiência

Configurações de TTL recomendadas

Para um desempenho ideal de SSAI, configure suas políticas de cache de CDN com estas configurações específicas de TTL:

Definições de configuração SSAI TTL

Tipo de conteúdo	Configuração TTL	Valor recomendado
Segmentos de anúncios	TTL mínimo	1 segundo
Segmentos de anúncios	TTL máximo	86400 segundos (24 horas)
Segmentos de anúncios	TTL padrão	86400 segundos (24 horas)
Segmentos de conteúdo	TTL mínimo	1 segundo
Segmentos de conteúdo	TTL máximo	86400 segundos (24 horas)
Segmentos de conteúdo	TTL padrão	86400 segundos (24 horas)

Essas configurações garantem:

- TTL mínimo de 1 segundo: permite a rápida invalidação do cache quando necessário, evitando solicitações de origem excessivas
- TTL máximo de 24 horas: equilibra a eficiência do cache com os requisitos de atualização do conteúdo
- TTL padrão de 24 horas: fornece armazenamento em cache ideal para segmentos que não têm cabeçalhos de controle de cache explícitos

Cache de inserção de anúncios guiada por servidor (SGAI)

A inserção de anúncios guiada pelo servidor (SGAI) permite o armazenamento eficiente em cache de CDN por meio de manifestos de mídia armazenáveis em cache que usam padrões de URL

previsíveis. Esta seção se concentra nos requisitos de configuração específicos da CDN para um desempenho ideal do SGAI.

Configuração de cache CDN para SGAI

Configure sua CDN com esses comportamentos de cache específicos do SGAI:

Configurações de cache do SGAI CDN

Tipo de conteúdo	TTL	Padrão de caminho	Elementos principais do cache
Playlists multivariantes SGAI (não armazenam em cache)	0 segundos (não armazene em cache)	/v1/master/ *	Caminho do URL + parâmetros de consulta selecionados
Playlists de mídia SGAI	1-4 segundos (comprimento de meio segmento)	/v1/i-media/ *	Caminho do URL + parâmetros de consulta selecionados
Respostas da lista de ativos (não armazenam em cache)	0 segundos (não armazene em cache)	/v1/intersticiais/ *	Caminho do URL + todos os parâmetros de consulta
Segmentos de anúncios	Mais de 24 horas	Caminhos específicos para anúncios	Somente caminho de URL

Configuração do comportamento do cache

Configure comportamentos de cache dedicados para o conteúdo do SGAI:

- Comportamento do manifesto SGAI - Crie um comportamento de cache para /v1/i-media/* caminhos com TTL de 1 a 4 segundos

- Comportamento da lista de ativos - Crie um comportamento de cache para `/v1/interstitials/*` caminhos com TTL de 0 segundo
- Manipulação de parâmetros de consulta - inclua somente parâmetros de segmentação essenciais nas chaves de cache para maximizar a eficiência do cache
- Cabeçalhos de solicitação de origem - encaminhe os cabeçalhos necessários para a segmentação de anúncios, mantendo a capacidade de armazenamento em cache

Cache de montagem de canais

Para fluxos de trabalho de montagem de canais, os manifestos podem ser armazenados em cache por curtos períodos, enquanto os segmentos devem ser armazenados em cache de forma agressiva:

Configurações de cache CDN de montagem de canais

Tipo de conteúdo	VOD TTL	TTL ao vivo	Padrão de caminho	Elementos principais do cache
Playlists multivariantes	5-30 minutos	5-10 segundos	Caminhos específicos do canal	Caminho do URL + todos os parâmetros de consulta
Playlists de mídia	5-30 minutos	2-5 segundos	Caminhos específicos do canal	Caminho do URL + todos os parâmetros de consulta
TRAVESSIA MPDs	5-30 minutos	5-10 segundos	Caminhos específicos do canal	Caminho do URL + todos os parâmetros de consulta
Segmentos de conteúdo	Mais de 24 horas	5-15 minutos	Caminhos específicos de conteúdo	Somente caminho de URL
Segmentos de anúncios	Mais de 24 horas	Mais de 24 horas	Caminhos específicos para anúncios	Somente caminho de URL

- Defina valores curtos de TTL para manifestos para garantir que os espectadores recebam a programação up-to-date

- Configure valores de TTL mais longos para segmentos de conteúdo para maximizar a eficiência do cache
- Configure comportamentos de cache que incluam parâmetros de mudança de horário na chave de cache se você oferecer suporte à visualização com mudança de horário
- Inclua parâmetros de consulta na chave de cache para lidar adequadamente com solicitações de visualização com mudança de horário

Para obter as configurações detalhadas do TTL e as melhores práticas, consulte [the section called “Otimização do cache”](#).

Cache combinado de SSAI e montagem de canais

Ao implementar a montagem de canais e o SSAI, garanta que sua estratégia de armazenamento em cache seja consistente em ambos os serviços para evitar conflitos e otimizar o desempenho:

Comparação de configurações de cache de fluxo de trabalho combinado

Tipo de conteúdo	Montagem do canal	SSAI	Recomendação combinada
Manifestos VOD	5-30 minutos	0 segundos	(use uma configuração separada)
Manifestos ao vivo	2-10 segundos	0 segundos	(use uma configuração separada)
Manifestos SGAI VOD	5-30 minutos	5-30 minutos	5-30 minutos (manifestos armazenáveis em cache)
Manifestos do SGAI Live	2-4 segundos	2-4 segundos	2 a 4 segundos (manifestos armazenáveis em cache)
Segmentos de conteúdo	Mais de 24 horas	Mais de 24 horas	Mais de 24 horas (consistente)
Segmentos de anúncios	Mais de 24 horas	Mais de 24 horas	Mais de 24 horas (consistente)

Essa configuração maximiza a eficiência do cache e garante que os espectadores recebam up-to-date manifestos para inserção de anúncios personalizados.

Otimização de roteamento de solicitações para CDN e integrações MediaTailor

Implemente essas otimizações de roteamento para todas AWS Elemental MediaTailor as integrações de CDN:

- Crie comportamentos de cache separados para solicitações de manifesto e segmento
- Configurar políticas de solicitação de origem para controlar o encaminhamento de cabeçalhos
- Configure mecanismos adequados de tratamento de erros e failover
- Implemente escudos de origem, se disponíveis em sua CDN, para reduzir a carga de origem
- Implemente o colapso de solicitações no nível da CDN para lidar com solicitações simultâneas com eficiência

Benchmarks de desempenho para CDN e integrações MediaTailor

Ao otimizar sua integração de AWS Elemental MediaTailor CDN, busque esses benchmarks de desempenho. Essas metas se aplicam a todas as MediaTailor implementações, incluindo SSAI, montagem de canais e fluxos de trabalho combinados:

Metas de taxa de acertos do cache

Segmentos de conteúdo: maior que 95% de taxa de acertos do cache

Segmentos de anúncios: taxa de acerto de cache superior a 90%

Manifestos: Não aplicável para SSAI (não devem ser armazenados em cache para inserção de anúncios personalizados); mais de 85% para montagem de canais

Benchmarks de latência

Latência da solicitação de manifesto: menos de 100 ms (P95)

Entrega do segmento de conteúdo: menos de 50 ms (P95)

Entrega do segmento de anúncios: menos de 75 ms (P95)

End-to-end tempo de inicialização: menos de 2 segundos

Métricas de carga de origem

Solicitações de origem por espectador: menos de 0,1 solicitações por minuto por espectador

Largura de banda de origem por espectador: menos de 5% da largura de banda total do espectador

Metas de taxa de erro

Erros manifestos: menos de 0,1%

Erros de segmento: menos de 0,01%

Rebuffer relatado pelo jogador: menos de 1%

Benchmarks de escalabilidade

Support para 10 vezes o tráfego normal durante eventos de pico sem degradação

Capacidade de lidar com mais de 1000 solicitações por segundo por canal

Use CloudWatch as métricas da Amazon para monitorar esses indicadores de desempenho. Para obter instruções detalhadas de monitoramento, consulte [Configurar ferramentas de monitoramento](#).

Técnicas avançadas de otimização para CDN e integrações MediaTailor

Depois de implementar as otimizações básicas de cache e roteamento, considere estas técnicas avançadas para aprimorar ainda mais o desempenho:

Implementação do Origin Shield

O Origin Shield adiciona uma camada de cache entre os pontos de borda da CDN e seu servidor de origem, reduzindo a carga de origem e melhorando o desempenho:

- Ative o Origin Shield para conteúdo de alto tráfego e transmissão ao vivo
- Escolha locais do Origin Shield próximos às suas MediaTailor regiões
- Monitore as taxas de acerto do cache do Origin Shield e ajuste conforme necessário
- Considere vários locais do Origin Shield para implantações globais

Otimização da compressão de conteúdo

Otimize a compactação de conteúdo para reduzir a largura de banda e melhorar a velocidade de entrega:

- Ativar a compactação gzip para arquivos de manifesto
- Configurar o encaminhamento do cabeçalho Accept-Encoding para compactação do manifesto MediaTailor
- Use a compressão Brotli onde houver suporte para economia adicional de largura de banda
- Evite compactar segmentos de vídeo já compactados

Estratégias de otimização regional

Otimize o desempenho para públicos globais por meio de estratégias regionais:

- Implemente MediaTailor configurações em várias regiões para públicos globais
- Use o roteamento geográfico para direcionar os espectadores para a região mais próxima MediaTailor
- Configure o failover regional para alta disponibilidade
- Monitore as métricas de desempenho regional separadamente

Considerações de arquitetura para CDN e integrações MediaTailor

Posicione sua rede de distribuição de conteúdo (CDN) corretamente em sua arquitetura para garantir desempenho e confiabilidade ideais com AWS Elemental MediaTailor. A arquitetura recomendada coloca a CDN entre os visualizadores e MediaTailor não entre MediaTailor sua origem.

Para obter diagramas detalhados de arquitetura e explicações do fluxo de trabalho, consulte os tópicos a seguir.

- [Inserção de anúncios com CDN](#) para diagramas de arquitetura de inserção de anúncios e fluxo de trabalho detalhado
- [Entenda a arquitetura CDN](#) para diagramas de arquitetura de montagem de canais e fluxo de trabalho

Posicione sua CDN corretamente em sua arquitetura:

1. Coloque seu CDN entre jogadores e MediaTailor (não entre MediaTailor e sua origem).

Essa arquitetura permite que sua CDN armazene segmentos de anúncios e segmentos de conteúdo. Ao mesmo tempo, MediaTailor pode gerar manifestos personalizados para cada espectador.

2. Crie comportamentos de cache separados para diferentes tipos de solicitação:

- Solicitações de manifesto (sem armazenamento em cache)
- Segmentos de conteúdo (TTL mais longo)
- Segmentos de anúncios (TTL mais longo)

3. Configure o tratamento adequado de erros:

- Configure o cache negativo (armazenando temporariamente as respostas de erro) para evitar sobrecarregar sua origem com solicitações repetidas durante interrupções no serviço. O armazenamento em cache negativo significa que a CDN armazenará temporariamente as respostas de erro (como erros 404 ou 500) para evitar solicitações repetidas de conteúdo que não existe ou está temporariamente indisponível.
- Configure os códigos de resposta de erro apropriados e o comportamento de repetição

4. Implemente o armazenamento em cache intermediário (escudo de origem):

O Origin Shield é um recurso que cria uma camada adicional de cache entre os pontos de presença da CDN e seu servidor de origem. Isso reduz o número de solicitações redundantes que chegam ao seu servidor de origem.

- Configure uma camada de cache intermediária entre as bordas e sua origem
- Reduza o número de solicitações redundantes para sua origem durante falhas de cache
- Melhore as taxas de acerto do cache em sua infraestrutura de CDN

Recursos avançados de CDN para MediaTailor

Depois de implementar a configuração básica da rede de distribuição de conteúdo (CDN), explore esses recursos avançados para aprimorar ainda mais o desempenho e a confiabilidade da sua plataforma de AWS Elemental MediaTailor streaming.

Roteamento com reconhecimento de qualidade de mídia (MQAR)

O MQAR é um CloudFront recurso da Amazon que seleciona automaticamente a fonte de conteúdo da mais alta qualidade com base em métricas de desempenho da rede em tempo real. Em vez de usar um servidor de origem fixa, o MQAR roteia dinamicamente as solicitações para a

origem com melhor desempenho com base em fatores como latência e taxa de transferência. Isso ajuda a garantir que os espectadores recebam a transmissão da mais alta qualidade possível, mesmo durante flutuações na rede.

Se você usa a Amazon CloudFront, implemente o MQAR para selecionar automaticamente a fonte de conteúdo da mais alta qualidade com base em métricas em tempo real. Para obter detalhes, consulte [Integração do CloudFront](#) na seção de CloudFront integração.

Filtragem de manifestos

A filtragem de manifestos permite que você personalize quais representações (versões de qualidade diferentes do mesmo conteúdo) estão incluídas nos manifestos que MediaTailor são entregues aos espectadores. A filtragem ajuda a otimizar o uso da largura de banda removendo representações que não são apropriadas para determinados dispositivos ou condições de rede. Por exemplo, você pode remover representações em 4K para dispositivos móveis ou conexões de baixa largura de banda.

Para obter informações detalhadas sobre a implementação da filtragem de manifestos com AWS Elemental MediaPackage, consulte [MediaPackage Integração CDN](#).

Estratégia multi-CDN

Uma estratégia de várias CDN usa vários provedores de CDN simultaneamente para melhorar a confiabilidade e o desempenho. Se uma CDN tiver problemas, o tráfego pode ser transferido automaticamente para outro provedor. Essa abordagem é particularmente valiosa para eventos ao vivo de alto perfil, nos quais a confiabilidade é fundamental.

Para obter informações sobre como implementar uma estratégia de várias CDN com MediaTailor, consulte [Planeje a integração com a CDN](#).

Monitore as operações e o desempenho da MediaTailor CDN

O monitoramento eficaz de sua AWS Elemental MediaTailor integração com a rede de distribuição de conteúdo (CDN) garante a entrega confiável de conteúdo, desempenho ideal e detecção rápida de problemas. Essa abordagem de monitoramento se aplica a todas as MediaTailor implementações, incluindo inserção de anúncios no lado do servidor (SSAI), montagem de canais e fluxos de trabalho combinados.

O monitoramento de sua integração com a CDN permite que você:

- Detecte e resolva problemas antes que eles afetem os espectadores
- Acompanhe os principais indicadores de desempenho e mantenha a qualidade do serviço
- Mantenha o desempenho ideal do cache e reduza a carga de origem
- Garanta que as taxas de sucesso de inserção de anúncios atendam aos requisitos de negócios

Para solucionar problemas relacionados a parâmetros que possam aparecer nos dados de monitoramento, consulte [MediaTailor guia de solução de problemas de parâmetros](#). Para obter informações sobre como monitorar o uso dos parâmetros de consulta, consulte [MediaTailor parâmetros de consulta do manifesto](#).

Métricas essenciais de desempenho da CDN

Acompanhe essas métricas principais para avaliar a eficácia da sua CDN com MediaTailor implementações:

Taxa de acertos do cache

O que ele mede: a porcentagem de solicitações atendidas do cache CDN versus da origem.

Valores-alvo:

- Segmentos de conteúdo: taxa de acerto de cache de 95% ou mais
- Segmentos de anúncios: taxa de acerto de cache de 90% ou mais
- Manifestos: varia de acordo com a implementação (manifestos personalizados não devem ser armazenados em cache)

Por que isso importa: taxas de acerto de cache mais altas reduzem a carga de origem, melhoram os tempos de resposta e reduzem os custos de largura de banda.

Volume de solicitações de origem

O que ele mede: o número de solicitações que chegam aos seus servidores de MediaTailor origem.

Padrão alvo: deve permanecer baixo e estável, com picos ocasionais de falhas de cache ou novos conteúdos.

Por que isso é importante: altos volumes de solicitações de origem indicam ineficiências no armazenamento em cache e podem afetar o desempenho. MediaTailor

Latência de resposta

O que ele mede: a rapidez com que o CDN responde às solicitações do espectador.

Valores-alvo:

- Conteúdo em cache: Menos de 100 ms (P95)
- Solicitações de origem: menos de 500 ms (P95)
- Solicitações de manifesto: menos de 100 ms (P95)
- Solicitações de segmento: menos de 50 ms (P95)

Por que isso é importante: a baixa latência garante uma reprodução suave e uma boa experiência para o espectador.

Taxas de erro

O que ele mede: a porcentagem de solicitações que resultam em erros de HTTP.

Valores-alvo:

- 4xx erros: menos de 0,1% do total de solicitações
- 5xx erros: menos de 0,01% do total de solicitações
- Erros de origem: menos de 0,05% das solicitações de origem

Por que isso importa: altas taxas de erro indicam problemas de configuração ou problemas de serviço que afetam a experiência do espectador.

MediaTailor métricas específicas

Monitore essas MediaTailor métricas junto com as métricas da CDN para obter visibilidade completa:

Taxas de preenchimento de anúncios

Principais métricas: `Avail.FillRate` e `AdDecisionServer.FillRate`

Valores alvo: acima de 90% para ambas as métricas

Por que isso é importante: afeta diretamente a receita publicitária e a experiência do espectador

Desempenho de geração de manifestos

Principais métricas: `GetManifest.Latency` e `GetManifest.Errors`

Valores alvo: latência abaixo de 200 ms, taxa de erro abaixo de 1%

Por que isso importa: afeta o tempo de inicialização e a confiabilidade da reprodução

Integridade do servidor de decisões de anúncios

Principais métricas: `AdDecisionServer.Latency`, `AdDecisionServer.Errors`, e `AdDecisionServer.Timeouts`

Valores alvo: latência abaixo de 1000 ms, taxa de erro abaixo de 5%, tempos limite mínimos

Por que isso é importante: o desempenho do ADS afeta diretamente o sucesso da inserção de anúncios

Configurar ferramentas de monitoramento

Configure essas ferramentas para monitorar sua integração com MediaTailor a CDN de forma eficaz:

CloudWatch Integração com a Amazon

A Amazon CloudWatch fornece a base para monitorar sua integração com MediaTailor a CDN:

MediaTailor métricas

MediaTailor publica automaticamente métricas CloudWatch que rastreiam solicitações, respostas e erros. As principais métricas incluem:

- `RequestCount`: Número total de solicitações para MediaTailor
- `ResponseTime`: latência de MediaTailor resposta
- `4xxErrorCount` e `5xxErrorCount`: Rastreamento de erros

Para obter uma lista completa de MediaTailor métricas, consulte [Monitoramento AWS Elemental MediaTailor com CloudWatch métricas da Amazon](#).

Métricas de CDN

Habilite a coleta de métricas de CDN CloudWatch incluindo:

- Taxas de acerto do cache para diferentes tipos de conteúdo
- Contagens de solicitações e tempos de resposta do Origin
- Taxas de erro por código de status

Para métricas CloudFront específicas, ative o monitoramento detalhado em suas configurações de distribuição.

Configuração de painel

Crie painéis que forneçam visibilidade do seu desempenho MediaTailor e do CDN:

1. Crie um painel unificado: combine MediaTailor métricas de CDN em um único CloudWatch painel para visibilidade completa.
2. Organize por fluxo de trabalho: agrupe métricas por tipo de implementação (SSAI, montagem de canais ou fluxos de trabalho combinados).
3. Inclua indicadores-chave de desempenho:
 - Tendências da taxa de acerto do cache ao longo do
 - Percentis de latência de resposta (P50, P95, P99)
 - Tendências e picos da taxa de erro

Para obter orientações detalhadas sobre a criação de painéis, consulte [CloudWatchpainéis](#).

Configuração de alertas

Configure alertas para detectar problemas antes que eles afetem os espectadores:

Alertas críticos

Configure alertas imediatos para problemas graves:

- Altas taxas de erro: alerta quando os erros 5xx excederem 0,1% das solicitações em 5 minutos
- Problemas no servidor de origem: alerta quando o tempo de resposta de origem exceder 2 segundos
- Diminuição da taxa de acertos do cache: alerta quando a taxa de acertos do cache cair abaixo de 70% para manifestos ou 85% para segmentos

Alertas de aviso

Configure alertas de aviso antecipado para degradação do desempenho:

- Aumento da latência: alerta quando o tempo de resposta do P95 exceder 200 ms
- Diminuição da eficiência do cache: alerta quando a taxa de acertos do cache cair abaixo de 90% nos segmentos

Lista de verificação de implementação

Use essa lista de verificação para garantir uma cobertura abrangente de monitoramento:

1. Coleção de métricas:

- ✓ MediaTailor métricas habilitadas em CloudWatch
- ✓ Monitoramento detalhado da CDN ativado

2. Configuração do painel:

- ✓ CloudWatch Painel unificado criado
- ✓ Principais métricas visualizadas com intervalos de tempo apropriados
- ✓ Painel compartilhado com equipes relevantes

3. Configuração de alertas:

- ✓ Alertas críticos configurados com notificação imediata
- ✓ Alertas de aviso configurados para detecção precoce
- ✓ Procedimentos de escalonamento de alertas documentados

4. Procedimentos operacionais:

- ✓ Procedimentos de resposta a incidentes documentados
- ✓ Cronograma de revisão regular estabelecido
- ✓ Treinamento de equipe concluído

Tópicos relacionados

Para obter orientação adicional sobre cenários de monitoramento específicos:

- Otimização de desempenho: para técnicas de otimização detalhadas com base em dados de monitoramento, consulte [Otimização do desempenho da CDN](#).
- Solução de problemas: para obter procedimentos detalhados de solução de problemas usando dados de monitoramento, consulte a documentação de solução de problemas específica do fluxo de trabalho.
- Análise de registros: para análise e monitoramento abrangentes de registros, consulte [Análise de log de integração de CDN e referência de código de erro para MediaTailor](#).

Teste e validação para CDN e integrações MediaTailor

AWS Elemental MediaTailor a integração da rede de distribuição de conteúdo (CDN) exige testes completos para garantir a entrega confiável de anúncios e o desempenho ideal. Os testes sistemáticos ajudam a identificar problemas de integração antes que eles afetem os espectadores. Use essa abordagem abrangente para validar sua rede de distribuição de conteúdo e a MediaTailor integração em diferentes cenários, configurações e condições de carga.

Tópicos relacionados:

- Para configuração operacional, incluindo monitoramento e registro, consulte [Monitoramento de CDN](#)
- Para solucionar problemas quando ocorrerem problemas, consulte [Solucionar problemas de integração com a CDN](#)
- Para obter orientação sobre otimização de desempenho, consulte [Guia de otimização de desempenho para CDN e integrações MediaTailor](#)

Tópicos

- [Pré-requisitos de teste e configuração para CDN e integrações MediaTailor](#)
- [Metodologia de teste sistemático para CDN e integrações MediaTailor](#)
- [Lista de verificação de testes de pré-implantação para CDN e integrações MediaTailor](#)
- [Referência de ferramentas e utilitários de teste para CDN e integrações MediaTailor](#)

Pré-requisitos de teste e configuração para CDN e integrações MediaTailor

AWS Elemental MediaTailor o teste de integração da rede de distribuição de conteúdo (CDN) requer preparação adequada e alocação de recursos. Antes de iniciar o teste sistemático, verifique se você tem os recursos e as ferramentas necessários.

Recursos necessários:

- Teste a distribuição de CDN configurada para espelhar as configurações de produção
- Teste a MediaTailor configuração com conteúdo e fontes de anúncios conhecidos
- Teste conteúdo com características previsíveis (duração, formato, marcadores de intervalo publicitário)

- Teste o servidor de decisão de anúncios ou simule respostas do ADS
- Vários dispositivos de teste e tipos de reprodutores

Ferramentas de teste:

- `curl` para teste de solicitação HTTP
- `ffprobe` para validação de manifesto HLS
- `mp4box` para validação do manifesto DASH
- Ferramentas de desenvolvimento de navegadores para análise de rede
- Players de vídeo para end-to-end testes

Metodologia de teste sistemático para CDN e integrações MediaTailor

AWS Elemental MediaTailor os testes de integração da rede de distribuição de conteúdo (CDN) devem seguir uma abordagem sistemática e em fases para garantir uma cobertura abrangente. Siga essa abordagem estruturada para testar de forma abrangente sua rede de distribuição de conteúdo e MediaTailor integração. Cada fase se baseia na anterior para isolar possíveis problemas.

Para obter orientação adicional sobre abordagens de testes sistemáticos, consulte [Testes de confiabilidade](#) no AWS Well-Architected Framework.

Fase 1: Teste a MediaTailor conectividade direta

Comece testando a MediaTailor funcionalidade sem o envolvimento da CDN para estabelecer uma linha de base.

1. Teste as solicitações do manifesto diretamente nos MediaTailor endpoints:

- Teste as solicitações de playlists multivariantes do HLS: `curl -v "https://your-emt-endpoint.mediatailor.region.amazonaws.com/v1/master/hls/config-name/master.m3u8"`
- Teste as solicitações de MPD do DASH: `curl -v "https://your-emt-endpoint.mediatailor.region.amazonaws.com/v1/dash/config-name/manifest.mpd"`
- Verifique se as respostas do manifesto contêm os marcadores de interrupção de anúncio esperados
- Verifique se o segmento URLs aponta para as origens corretas

2. Verifique se a inserção do anúncio está funcionando corretamente:

- Teste com diferentes parâmetros de segmentação de anúncios
- Verifique se os segmentos de anúncios estão devidamente transcodificados e disponíveis
- Verifique o tempo e a duração do intervalo publicitário
- Teste o comportamento alternativo quando os anúncios não estão disponíveis

3. Avalie o desempenho básico:

- Registrar os tempos de resposta da solicitação de manifesto
- Avalie os tempos de resposta do servidor de decisões publicitárias
- Criação e gerenciamento de sessões de teste

CrITÉRIOS de sucesso: todas as MediaTailor solicitações diretas retornam respostas HTTP 200 com manifestos formatados corretamente contendo o conteúdo esperado do anúncio.

Fase 2: Teste a integração básica do CDN

Adicione CDN ao caminho da solicitação e teste a funcionalidade básica.

1. Teste as solicitações de manifesto por meio da CDN:

- Configure a CDN com regras de roteamento simples
- Teste solicitações de manifesto por meio de endpoints CDN
- Verifique se o CDN encaminha corretamente as solicitações para MediaTailor
- Verifique se as respostas do manifesto não estão armazenadas em cache (TTL = 0)

2. Roteamento do segmento de teste:

- Verifique se os segmentos de conteúdo são roteados para o servidor de origem
- Verifique a rota dos segmentos de MediaTailor anúncios para o armazenamento de anúncios
- Teste o comportamento do cache do segmento

3. Compare o CDN com o desempenho direto:

- Avalie as diferenças de tempo de resposta
- Verifique se há diferenças de conteúdo nas respostas
- Verifique se o tratamento de erros funciona corretamente

CrITÉRIOS de sucesso: a CDN deve fazer proxy de solicitações MediaTailor e servidores de origem com sucesso, com impacto mínimo no desempenho.

Fase 3: Encaminhamento de parâmetros de consulta de teste

Adicione o encaminhamento de parâmetros de consulta e teste a personalização de anúncios.

1. Configure o encaminhamento de parâmetros de consulta na CDN:

- Ativar o encaminhamento de todos os parâmetros de consulta para MediaTailor
- Inicialização da sessão de teste (o ID da sessão é gerado automaticamente MediaTailor na primeira solicitação)
- Teste com parâmetros de segmentação personalizados

2. Teste e personalização de anúncios:

- Verifique se parâmetros diferentes produzem respostas de anúncios diferentes
- Codificação de parâmetros de teste e caracteres especiais
- Verifique se os parâmetros foram passados corretamente para o ADS

3. Valide o gerenciamento da sessão:

- Criação e persistência da sessão de teste
- Verifique a consistência do ID da sessão em todas as solicitações
- Tratamento da expiração da sessão de teste

CrITÉRIOS DE SUCESSO: o conteúdo do anúncio varia de acordo com os parâmetros de consulta e as sessões são gerenciadas corretamente.

Fase 4: Encaminhamento do cabeçalho de teste

Adicione o encaminhamento de cabeçalho na CDN e teste a segmentação específica do dispositivo.

1. Configure o encaminhamento de cabeçalhos para todos os cabeçalhos. Para obter os requisitos mínimos, consulte [Cabeçalhos necessários para integração com MediaTailor CDN](#).

2. Teste a segmentação por dispositivos:

- Teste com diferentes cadeias de caracteres do User-Agent (celular, desktop, TV)
- Verifique as respostas de anúncios específicas do dispositivo
- Teste a segmentação geográfica com diferentes endereços IP

3. Valide o tratamento do CORS:

- Teste os cabeçalhos CORS para compatibilidade com reprodutores web
- Verifique se as solicitações de preflight OPTIONS funcionam corretamente

- Teste em diferentes domínios

Critérios de sucesso: a segmentação geográfica e por dispositivo deve funcionar corretamente, e os players da web não devem encontrar erros de CORS.

Fase 5: Teste abrangente de cenários

Teste em vários cenários para garantir uma operação robusta.

1. Teste com diferentes tipos de jogadores:

- Jogadores HLS.js em navegadores da web
- Jogadores Video.js com suporte para HLS e DASH
- Jogadores nativos em dispositivos móveis
- Smart TV e tocadores de set-top box

2. Teste em diferentes dispositivos e plataformas:

- Dispositivos móveis (iOS, Android)
- Navegadores de desktop (Chrome, Firefox, Safari, Edge)
- Dispositivos inteligentes TVs e de streaming
- Diferentes versões do sistema operacional

3. Teste diferentes tipos de conteúdo:

- Conteúdo de transmissão ao vivo
- Conteúdo de vídeo sob demanda (VOD)
- Diferentes formatos de vídeo e taxas de bits
- Conteúdo com diferentes padrões de interrupção de anúncios

4. Teste cenários de segmentação de anúncios:

- Diferentes parâmetros de segmentação demográfica
- Segmentação geográfica em diferentes regiões
- Segmentação com base no tempo (diferentes horários do dia)
- Parâmetros de lógica de negócios personalizados

Critérios de sucesso: todas as combinações de jogadores e dispositivos devem funcionar corretamente com a segmentação de anúncios adequada.

Fase 6: Teste de carga e desempenho

Valide o desempenho sob condições de carga realistas.

Important

Antes do teste de carga, entre em contato com o [AWS Support](#): Antes de realizar testes de carga e desempenho, crie um ticket de AWS suporte para notificar a equipe de MediaTailor serviço sobre seus testes planejados. Isso garante:

- O serviço está preparado para os níveis de carga esperados
- A capacidade apropriada está disponível durante sua janela de teste
- Seu teste não será confundido com um incidente de produção
- Você recebe orientação sobre as melhores práticas e limitações de testes

Inclua em seu ticket de suporte: usuários simultâneos esperados, duração do teste, regiões geográficas e quaisquer cenários específicos que você planeja testar.

1. Teste cenários de usuários simultâneos:

- Simule vários visualizadores simultâneos
- Teste o dimensionamento da CDN e o desempenho do cache
- Monitore o desempenho do servidor de origem sob carga
- Verifique se o MediaTailor pode lidar com sessões simultâneas

2. Avalie as métricas de desempenho:

- Monitore os tempos de resposta sob carga
- Verifique se as taxas de acerto do cache atendem às expectativas (> 80% para conteúdo popular)
- Meça o tempo até o primeiro quadro em diferentes cenários
- Rastreie as taxas de erro durante o pico de carga

3. Teste cenários de failover:

- Teste o comportamento quando o ADS não está disponível
- Teste o failover do servidor de origem
- Verifique o tratamento e a recuperação de erros

- Teste o failover da localização periférica da CDN

CrITÉRIOS DE SUCESSO: O sistema deve manter um desempenho aceitável sob a carga esperada, com degradação suave durante falhas. Certifique-se de entrar em contato com o [AWS Support](#) e eles aprovarão seu plano de teste de carga antes da execução.

Lista de verificação de testes de pré-implantação para CDN e integrações MediaTailor

AWS Elemental MediaTailor a integração da rede de distribuição de conteúdo (CDN) deve passar por testes abrangentes antes da implantação na produção. Use essa lista de verificação antes de implantar alterações de configuração na produção.

Funcionalidade básica:

- ☐ Solicitações de manifesto retornam respostas HTTP 200
- ☐ Segmentos de conteúdo são carregados corretamente
- ☐ Segmentos de anúncios são carregados corretamente
- ☐ Os intervalos publicitários aparecem nos horários esperados
- ☐ A reprodução transita suavemente entre conteúdo e anúncios

Validação da configuração:

- ☐ Os parâmetros de consulta são encaminhados corretamente
- ☐ Os cabeçalhos necessários são encaminhados corretamente
- ☐ O cache do manifesto está desativado (TTL = 0)
- ☐ O cache de segmentos está configurado adequadamente
- ☐ Os cabeçalhos CORS são configurados para reprodutores da web

Teste multiplataforma:

- ☐ Testado em dispositivos móveis
- ☐ Testado em navegadores de desktop
- ☐ Testado com diferentes tipos de jogadores

- ☐ Testado nos formatos HLS e DASH

Validação de desempenho:

- ☐ Ticket de AWS suporte criado para aprovação do teste de carga
- ☐ Os tempos de resposta atendem às metas de desempenho
- ☐ As taxas de acerto do cache são aceitáveis
- ☐ As taxas de erro estão dentro dos limites aceitáveis
- ☐ O monitoramento e o alerta estão configurados

Referência de ferramentas e utilitários de teste para CDN e integrações MediaTailor

AWS Elemental MediaTailor o teste de integração da rede de distribuição de conteúdo (CDN) requer várias ferramentas para validação e depuração abrangentes. Guia de referência para ferramentas comumente usadas em testes de rede e MediaTailor integração de distribuição de conteúdo.

curl- Teste de solicitação HTTP

Solicitações de manifesto de teste: `curl -v "https://your-cdn-domain.com/v1/master/hls/config/master.m3u8"`

Teste com cabeçalhos: `curl -H "User-Agent: TestAgent/1.0" "https://your-cdn-domain.com/..."`

Teste com parâmetros: `curl "https://your-cdn-domain.com/...?aws.sessionId=test123"`

ffprobe- Validação do manifesto HLS

Valide a sintaxe do HLS: `ffprobe -v quiet -print_format json -show_format "https://your-cdn-domain.com/master.m3u8"`

Verifique as informações do segmento: `ffprobe -v quiet -show_entries packet=pts_time "segment.ts"`

mp4box- Validação do manifesto DASH

Valide o DASH MPD: `mp4box -info "https://your-cdn-domain.com/manifest.mpd"`

Verifique a temporização do segmento: `mp4box -info segment.m4s`

Ferramentas para desenvolvedores de navegadores

Monitore solicitações de rede na guia Rede

Verifique se há erros de CORS na guia Console

Inspecione os cabeçalhos request/response

Analise métricas de tempo e desempenho

Recursos adicionais:

- [Solução de problemas CloudFront](#) - Guia abrangente de solução de problemas de CDN
- [Aumentar a taxa de acertos do CloudFront cache](#) - Guia de otimização de desempenho
- [Monitore AWS recursos](#) — Melhores práticas de monitoramento de desempenho

Solucionar problemas com a integração com MediaTailor a CDN

Este guia abrangente de solução de problemas aborda problemas comuns de integração da rede de distribuição de conteúdo (CDN) em todas as AWS Elemental MediaTailor implementações, incluindo inserção de anúncios no lado do servidor (SSAI), montagem de canais e integração. AWS Elemental MediaPackage Quando sua CDN e sua MediaTailor integração tiverem problemas, use essa abordagem sistemática de diagnóstico para identificar rapidamente as causas-raiz e implementar soluções validadas.

Este guia se aplica a todas as integrações de MediaTailor CDN, independentemente do seu fluxo de trabalho específico. Para problemas específicos de serviços ou fluxos de trabalho específicos, consulte as seções de solução de problemas relacionadas mencionadas no final deste guia.

Antes de começar: tenha esses itens prontos para uma solução eficiente de problemas:

- Amostra de reprodução URLs que demonstra o problema
- Registros de acesso à CDN do período em que os problemas ocorreram
- MediaTailor nome da configuração e AWS região
- Tipo e versão do player (por exemplo, HLS.js 1.4.0, Video.js 8.0)

- Informações do dispositivo e do navegador onde ocorrem problemas

Tópicos relacionados:

- Para configuração operacional e preparação para solução de problemas, consulte [Solucionar problemas de integração com a CDN](#)
- Para análise de log e referência de código de erro, consulte [Referência de análise de log de integração CDN](#)
- Para escalar e obter ajuda adicional, consulte [Obtenha suporte para integração de CDN](#)

Lista de verificação de diagnóstico para integrações MediaTailor de CDN

AWS Elemental MediaTailor problemas de integração da rede de distribuição de conteúdo (CDN) podem se manifestar de várias maneiras. Use essa lista de verificação para identificar rapidamente o tipo de problema que você está enfrentando:

1. O problema está afetando todos os espectadores ou espectadores específicos?
 - Todos os visualizadores → Provável problema de CDN ou MediaTailor configuração
 - Visualizadores específicos → Provável problema de personalização ou segmentação
2. Os manifestos estão sendo carregados corretamente?
 - Não → Problema de roteamento ou MediaTailor conectividade de CDN
 - Sim, mas conteúdo errado → Problema de cache ou personalização
3. Os segmentos estão carregando corretamente?
 - Falha nos segmentos de conteúdo → Problema de conectividade do Origin
 - Falha nos segmentos de anúncios → Problema na entrega ou transcodificação do anúncio
4. Os anúncios estão sendo inseridos corretamente?
 - Nenhum anúncio aparece → Verifique a conectividade e a configuração do ADS
 - Anúncios errados aparecem → Verifique os parâmetros de segmentação de anúncios e a personalização
 - Falha na reprodução dos anúncios → Verifique a transcodificação do anúncio e a disponibilidade do segmento

5. A reprodução é suave e ininterrupta?

- Problemas de buffer → Verifique o desempenho do cache CDN e os tempos de resposta de origem
- Erros de reprodução → Verifique a sintaxe do manifesto e a disponibilidade do segmento
- Problemas de transição de anúncios → Verifique o tempo de intervalo do anúncio e o alinhamento do segmento

6. Há códigos de erro ou mensagens específicos?

- Erros HTTP 4xx → Verifique o roteamento e a configuração do CDN
- Erros HTTP 5xx → Verifique a integridade do servidor e do MediaTailor serviço de origem
- Erros específicos do player → Verifique o formato do manifesto e a compatibilidade do player

Próximas etapas com base em seu diagnóstico:

Problemas de configuração do CDN

Para obter uma solução detalhada de problemas de roteamento e armazenamento em cache da CDN, consulte [Solucionar problemas com a integração com MediaTailor a CDN](#)

Problemas de manifesto e reprodução

Para a validação do manifesto e a solução de problemas de reprodução, consulte [Procedimentos de teste de integração CDN](#).

Problemas de inserção e segmentação de anúncios

Para solucionar problemas específicos do anúncio, incluindo conectividade e entrega de anúncios do ADS, consulte a documentação de solução de problemas específica do fluxo de trabalho.

Problemas de desempenho e monitoramento

Para análise de desempenho e configuração de monitoramento, consulte [Monitore as operações e o desempenho da MediaTailor CDN](#).

Análise de registros e códigos de erro

Para uma análise detalhada do registro e referência do código de erro, consulte [Análise de log de integração de CDN e referência de código de erro para MediaTailor](#).

Testes e validação

Para procedimentos de teste abrangentes, consulte [Teste e validação para CDN e integrações MediaTailor](#).

Se você precisar de assistência imediata ou não conseguir resolver o problema usando os recursos vinculados, consulte os procedimentos [Obtenha suporte e ajuda para solução de problemas para CDN e integrações MediaTailor](#) de escalonamento.

Procedimentos de teste de integração CDN

O teste adequado é essencial antes de implantar sua integração de MediaTailor CDN na produção. Esses procedimentos de teste ajudam a identificar problemas de configuração, problemas de desempenho e problemas de compatibilidade em diferentes dispositivos e plataformas.

Validação básica da integração

Execute esses testes fundamentais para verificar se sua integração com a CDN está funcionando corretamente:

1. Entrega do manifesto de teste:

- Solicite um manifesto por meio de sua CDN e verifique se ele retorna uma resposta válida
- Verifique se o manifesto contém o conteúdo esperado e os pontos de inserção do anúncio
- Verifique se o URLs manifesto usa seu domínio CDN, não a origem
- Valide a sintaxe do manifesto usando ferramentas de validação HLS ou DASH

2. Verifique a reescrita do URL:

- Verifique se o segmento de conteúdo URLs nos manifestos aponta para seu domínio CDN
- Verifique se o segmento de anúncios URLs aponta para seu domínio CDN
- Certifique-se de que todos os parentes URLs sejam resolvidos adequadamente

3. Teste a reprodução do conteúdo:

- Reproduza conteúdo por meio de um reprodutor de vídeo e verifique se a reprodução está fluida
- Verifique se o conteúdo e os anúncios são reproduzidos sem interrupção
- Verifique as transições adequadas entre conteúdo e anúncios
- Funcionalidade de busca e depuração de testes

4. Valide o roteamento CDN:

- Monitore os registros de acesso à CDN para garantir que as solicitações sejam roteadas corretamente
- Verifique se hit/miss os padrões de cache estão conforme o esperado

- ~~Verifique se as solicitações de origem ocorrem apenas em caso de falhas de cache~~

Procedimentos de teste avançados

Execute esses testes adicionais para uma validação abrangente:

1. Teste de compatibilidade entre plataformas:

- Teste em vários dispositivos (desktop, celular, tablet, smart TV)
- Verifique a compatibilidade em diferentes navegadores
- Teste com vários reprodutores de vídeo (HLS.js, Video.js, reprodutores nativos)
- Valide em diferentes sistemas operacionais

2. Teste de desempenho:

- Meça os tempos de resposta da solicitação de manifesto (meta: <100 ms para armazenamento em cache)
- Teste o desempenho do carregamento do segmento em diferentes taxas de bits
- Verifique se o tempo de inicialização atende às metas de desempenho
- Teste sob várias condições de rede

3. Validação do rastreamento de anúncios:

- Verifique se os beacons de rastreamento de anúncios foram acionados corretamente
- Verifique a precisão dos dados de análise de anúncios
- Rastreamento de impressão e conclusão de testes
- Valide a funcionalidade de clique

4. Teste de condição de erro:

- Teste o comportamento quando a origem está temporariamente indisponível
- Verifique o tratamento adequado de solicitações malformadas
- Teste cenários de failover de CDN
- Valide a clareza e a utilidade da mensagem de erro

Crie um ambiente de teste

Configure um ambiente de teste que espelhe sua configuração de produção para uma validação abrangente:

1. Configure distribuições CDN separadas para testes:

- Crie distribuições CDN de teste com os mesmos comportamentos de cache da produção

- Configure origens de teste que espelhem sua configuração de produção
 - Use nomes de domínio separados para evitar conflitos com o tráfego de produção
2. Crie MediaTailor configurações de teste:
 - Defina as configurações de reprodução de teste com as mesmas configurações da produção
 - Configurar endpoints do servidor de teste e decisão
 - Use conteúdo de anúncio de teste que corresponda aos seus formatos de anúncio de produção
 3. Implemente processos de teste sistemáticos:
 - Crie listas de verificação de teste para alterações de configuração
 - Documente os procedimentos de teste para sua equipe
 - Configure testes automatizados sempre que possível

Teste em vários cenários

Valide sua integração em diferentes cenários e condições para garantir uma cobertura abrangente:

1. Teste com vários tipos de jogadores e dispositivos:
 - Teste com diferentes reprodutores de vídeo (web, celular, TV conectada)
 - Valide em diferentes sistemas operacionais e navegadores
 - Teste em várias condições de rede e velocidades de conexão
2. Crie scripts de teste automatizados:
 - Automatize a validação da solicitação de manifesto
 - Crie scripts para testar cenários de inserção de anúncios
 - Implemente testes de desempenho para cenários de alto tráfego
3. Valide a segmentação e a personalização de anúncios:
 - Teste com diferentes perfis de usuário e parâmetros de segmentação
 - Validar a integração do servidor de decisões de anúncios
 - Teste cenários alternativos quando os anúncios não estão disponíveis

Ferramentas e técnicas de teste

Use essas ferramentas e técnicas para testes eficazes:

Ferramentas para desenvolvedores de navegadores

Use a guia Rede para inspecionar solicitações e respostas HTTP

Monitore o console em busca JavaScript de erros e avisos

Verifique os cabeçalhos de resposta e o comportamento do cache

Verifique as informações de tempo para análise de desempenho

Teste de linha de comando

Use curl para testar cabeçalhos específicos URLs e inspecionar:

```
curl -I "https://your-cdn-domain.com/path/to/manifest.m3u8"
```

Use o wget para baixar, testar e analisar o tempo.

Use ferramentas como httpie para testes HTTP mais legíveis

Teste de reprodutor de vídeo

Teste com implementações de vários jogadores

Use os modos de depuração do player para inspecionar o comportamento interno

Monitore eventos do player e retornos de chamada de erro

Valide o comportamento de comutação de taxa de bits adaptável

Análise e monitoramento de CDN

Monitore métricas de CDN em tempo real durante o teste

Revise os registros de acesso para ver os padrões de solicitação

Use ferramentas de teste específicas da CDN quando disponíveis

Configure alertas temporários para validação de testes

Para obter metodologias de teste abrangentes adicionais e abordagens de validação sistemática, consulte [Teste e validação para CDN e integrações MediaTailor](#).

Solucione os erros 404 do manifesto CDN para MediaTailor

AWS Elemental MediaTailor Os erros 404 do manifesto da rede de distribuição de conteúdo (CDN) são um problema comum de integração que impede o início da reprodução. Esta seção fornece step-by-step solução de problemas para falhas na entrega de manifestos.

Playlist multivariante, playlist de mídia ou solicitações de MPD retornam erros 404

Solução rápida (tente primeiro):

1. Verifique se o nome da MediaTailor configuração em seu URL corresponde exatamente (diferencia maiúsculas de minúsculas)
2. Teste o URL do manifesto diretamente MediaTailor sem CDN: `curl -v "https://your-
emt-endpoint.mediatailor.region.amazonaws.com/v1/master/hls/config-
name/master.m3u8"`
3. Se o teste direto funcionar, verifique as regras de roteamento da CDN para solicitações manifestas

Se a solução rápida não funcionar:

Sintomas: os players não conseguem iniciar a reprodução, as solicitações de manifesto retornam erros HTTP 404 nos registros do CDN.

Exemplos de mensagens de erro:

- Console do navegador: "Failed to load resource: the server responded with a status of 404 (Not Found)"
- Erro do jogador: "MANIFEST_LOAD_ERROR" ou "NETWORK_ERROR"
- Registros CDN: GET /v1/master/hls/example-config/master.m3u8 404

Resolução:

Verifique se suas regras de roteamento de CDN estão configuradas corretamente para encaminhar solicitações de playlists multivariantes, playlists de mídia e MPD para MediaTailor

Verifique se a MediaTailor configuração existe e está configurada corretamente.

Certifique-se de que seus padrões de comportamento de CDN correspondam aos caminhos de solicitação de manifesto esperados (por exemplo, *.m3u8, *.mpd).

Diagnostique problemas e erros de entrega de manifestos da CDN para MediaTailor

AWS Elemental MediaTailor problemas de entrega de manifestos na rede de distribuição de conteúdo (CDN) podem impedir a inserção e a reprodução adequadas de anúncios. Se os espectadores receberem anúncios incorretos ou inconsistentes em playlists multivariantes, playlists de mídia ou: MPDs

1. Verifique se há manifestos em cache:

- Verifique se as configurações de TTL estão definidas como 0 para todas as playlists multivariantes, playlists de mídia e caminhos MPD
- Confirme se sua CDN não está armazenando em cache playlists multivariantes, playlists de mídia ou apesar das configurações de TTL MPDs
- Verifique o status do cache nos registros do CDN - as solicitações de manifesto devem ser exibidas Miss ou RefreshHit não Hit

2. Verifique a configuração de roteamento CDN:

- Confirme se as solicitações do manifesto estão sendo roteadas para MediaTailor endpoints, não armazenadas em cache ou atendidas a partir da origem
- Verifique se os padrões de comportamento da CDN correspondem corretamente aos caminhos do manifesto (*.m3u8, *.mpd)
- Verifique se os parâmetros de consulta estão sendo encaminhados MediaTailor para personalização
- Teste o manifesto URLs diretamente MediaTailor para isolar problemas de CDN versus problemas de serviço

3. Verifique a configuração de encaminhamento do cabeçalho:

- Verifique se os cabeçalhos necessários estão sendo encaminhados (consulte) [Cabeçalhos necessários para integração com MediaTailor CDN](#)
- Confirme se o cabeçalho do User-Agent foi encaminhado para a segmentação de anúncios específica do dispositivo
- Verifique se o X-Forwarded-For cabeçalho foi encaminhado para segmentação geográfica
- Certifique-se de que o cabeçalho Accept-Encoding seja encaminhado para suporte à compressão

4. Valide o conteúdo e a estrutura do manifesto:

- Verifique se os manifestos contêm os marcadores de inserção de anúncios esperados (EXT-X-CUE-OUT/IN para HLS)
- Verifique se o segmento URLs nos manifestos usa seu domínio CDN, não domínios de origem
- Confirme se os segmentos do anúncio estão inseridos e acessíveis corretamente
- Teste a sintaxe do manifesto usando ferramentas de validação (ffprobe para HLS, mp4box para DASH)

5. Teste diferentes cenários:

- Teste com uma sessão diferente IDs para verificar se a personalização está funcionando
- Teste em diferentes localizações geográficas para verificar a segmentação geográfica
- Teste com diferentes sequências de caracteres do User-Agent para verificar a segmentação do dispositivo
- Compare respostas manifestas com e sem CDN para identificar diferenças

Recursos adicionais de solução de problemas:

- Para obter uma configuração detalhada do cache da CDN, consulte [Otimização de cache para CDN e integrações MediaTailor](#)
- Para uma configuração abrangente de roteamento CDN, consulte [Configure comportamentos de roteamento de CDN para MediaTailor](#)
- Para requisitos de encaminhamento de cabeçalho, consulte [Cabeçalhos necessários para integração com MediaTailor CDN](#)
- Para análise de registros e códigos de erro, consulte [Análise de log de integração de CDN e referência de código de erro para MediaTailor](#)
- Para procedimentos de teste e validação, consulte [Teste e validação para CDN e integrações MediaTailor](#)

CrITÉRIOS DE SUCESSO: quando resolvidos, os jogadores devem iniciar a reprodução normalmente e os anúncios devem aparecer conforme o esperado. As solicitações de manifesto devem retornar códigos de status HTTP 200 nos registros da CDN, e os manifestos devem conter conteúdo de anúncio devidamente personalizado.

Solucione problemas de entrega e carregamento de segmentos CDN para MediaTailor

AWS Elemental MediaTailor problemas de entrega de segmentos da rede de distribuição de conteúdo (CDN) podem causar interrupções no buffer e na reprodução. Se os jogadores não conseguirem carregar segmentos ou experimentar o buffer:

1. Verifique as regras de roteamento da CDN:

- Verifique se os segmentos de conteúdo estão sendo roteados para a origem correta
- Confirme se os segmentos de anúncios estão sendo roteados para o local correto de armazenamento de MediaTailor anúncios
- Verifique se as extensões de arquivo do segmento correspondem aos padrões de comportamento do seu CDN
- Verifique se o segmento URLs nos manifestos usa o domínio CDN correto
- Para obter instruções detalhadas sobre como configurar padrões de roteamento e caminhos de comportamento, consulte [Configure comportamentos de roteamento de CDN para MediaTailor](#)

2. Verifique a configuração do CORS:

- Para reprodutores da web, certifique-se de que seu CDN esteja passando ou configurando corretamente os cabeçalhos CORS
- Teste com ferramentas de desenvolvedor de navegadores para identificar erros relacionados ao CORS
- Verifique se as solicitações de preflight OPTIONS são tratadas corretamente

3. Teste a acessibilidade e o desempenho do segmento:

- Teste segmentos individuais URLs diretamente para verificar se eles estão acessíveis
- Verifique os tempos de resposta do segmento e identifique gargalos de desempenho
- Verifique se os tamanhos dos arquivos do segmento são apropriados para suas metas de largura de banda
- Teste o carregamento de segmentos de diferentes localizações geográficas

4. Valide o comportamento de armazenamento em cache da CDN para segmentos:

- Verifique se os segmentos de conteúdo têm configurações de TTL apropriadas (geralmente mais longas que os manifestos)
- Verifique se os segmentos de anúncios são armazenados em cache adequadamente com base

- Monitore as taxas de acerto do cache para conteúdo e segmentos de anúncios
 - Garanta que as chaves de cache não incluam parâmetros desnecessários que reduzam a eficiência do cache
5. Verifique a conectividade e a integridade do servidor de origem:
- Verifique se os servidores de origem estão respondendo corretamente às solicitações de segmento
 - Verifique a capacidade do servidor de origem e os tempos de resposta sob carga
 - Valide se os servidores de origem têm os arquivos de segmento esperados disponíveis
 - Teste cenários de failover de origem se várias origens estiverem configuradas
6. Solucione problemas específicos do segmento de anúncios:
- Verifique se os segmentos de anúncios estão devidamente transcodificados e estão disponíveis no MediaTailor
 - Verifique se os segmentos de anúncios foram URLs gerados corretamente nos manifestos
 - Teste o carregamento do segmento de anúncios com diferentes parâmetros de segmentação de anúncios
 - Monitore atrasos na transcodificação de anúncios que podem causar indisponibilidade do segmento
7. Valide a compatibilidade e o comportamento do jogador:
- Teste o carregamento do segmento com diferentes tipos e versões de players
 - Verifique as configurações do buffer do player e os padrões de solicitação de segmentos
 - Verifique o tratamento de erros do player para solicitações de segmentos com falha
 - Teste a comutação adaptativa de taxa de bits e a lógica de seleção de segmentos

Recursos adicionais de solução de problemas:

- Para configuração de roteamento e comportamento de CDN, consulte [Configure comportamentos de roteamento de CDN para MediaTailor](#)
- Para otimizar o cache do CDN, consulte [Otimização de cache para CDN e integrações MediaTailor](#)
- Para obter orientação sobre a configuração do CORS, consulte [Melhores práticas de segurança de integração de CDN para MediaTailor](#)
- Para monitoramento e análise de desempenho, consulte [Monitore as operações e o desempenho da MediaTailor CDN](#)

- Para procedimentos de teste abrangentes, consulte [Teste e validação para CDN e integrações MediaTailor](#)
- Para análise de registros e diagnóstico de erros, consulte [Análise de log de integração de CDN e referência de código de erro para MediaTailor](#)

CrITÉRIOS de sucesso: Quando resolvidos, os jogadores devem carregar segmentos sem problemas, sem interrupções no buffer. As solicitações de segmentos devem retornar códigos de status HTTP 200 com tempos de resposta adequados, e tanto o conteúdo quanto os segmentos de anúncios devem estar acessíveis e devidamente armazenados em cache.

Corrija problemas de gerenciamento e rastreamento de sessões de CDN para MediaTailor

AWS Elemental MediaTailor o gerenciamento de sessões da rede de distribuição de conteúdo (CDN) é fundamental para a personalização e o rastreamento adequados de anúncios. Se você encontrar erros relacionados à sessão ou comportamento inconsistente em todas as solicitações:

1. Verifique a consistência do ID da sessão:

- Verifique se o player mantém o mesmo ID de sessão em todas as solicitações de uma única sessão de reprodução
- Verifique os registros do CDN para confirmar se a sessão IDs está sendo encaminhada corretamente
- Certifique-se de que a sessão IDs esteja devidamente codificada em URL nos parâmetros de consulta
- Use CloudWatch registros para verificar a consistência do ID da sessão em todas as solicitações (consulte as etapas de validação abaixo)

2. Valide a inicialização da sessão:

- Confirme se a primeira solicitação de manifesto cria uma sessão com sucesso
- Verifique o encaminhamento correto dos parâmetros da sessão (por exemplo, `aws.sessionId`)
- Verifique a inicialização da sessão usando registros de depuração (consulte a configuração do registro de depuração abaixo)

3. Ative o registro de depuração para solucionar problemas detalhados da sessão:

- Para relatórios do lado do servidor: adicione `?aws.logMode=DEBUG` à sua solicitação de reprodução:

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

- Para relatórios do lado do cliente: inclua "logMode": "DEBUG" no corpo da solicitação de inicialização da sessão
- Importante: o DEBUG valor diferencia maiúsculas de minúsculas
- Máximo de 10 sessões de depuração ativas permitidas simultaneamente

4. Use consultas CloudWatch de registros para validar o comportamento da sessão:

- Verifique se a sessão de depuração está ativa:

```
fields @timestamp, @message  
| filter sessionId = "your-session-id-here"  
| 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
```

- Veja todos os eventos de uma sessão:

```
fields @timestamp, @message, eventType, mediaTailorPath  
| filter sessionId = "your-session-id-here"  
| sort @timestamp asc
```

- Verifique a geração do manifesto para uma sessão:

```
fields @timestamp, responseBody, @message  
| filter mediaTailorPath like "/v1/master/" and eventType = "GENERATED_MANIFEST"  
and sessionId = "your-session-id-here"
```

5. Encaminhamento de parâmetros da sessão de teste por meio de CDN:

- Teste solicitações de manifesto com parâmetros de sessão diretamente MediaTailor (ignorando a CDN)
- Compare o comportamento da sessão com e sem CDN para identificar problemas de encaminhamento
- Verifique se a configuração de encaminhamento de parâmetros de consulta CDN inclui parâmetros relacionados à sessão
- Verifique se o CDN não armazena em cache as respostas que deveriam ser específicas da sessão

Mensagens comuns de erro de sessão:

- `ConflictException(HTTP 409)` - Várias solicitações simultâneas de playlists para a mesma sessão. Solução: garanta que seu player solicite playlists uma de cada vez, de acordo com a especificação HLS
- `NotFoundException(HTTP 404)` - A sessão não está disponível ou a configuração não existe. Solução: verifique a validade da configuração e reinicialize a sessão
- `BadRequestException(HTTP 400)` - ID de sessão inválida ou solicitação formatada incorretamente. Solução: verifique a formatação da solicitação e a validade do ID da sessão

Recursos adicionais de solução de problemas:

- Para obter a configuração completa do registro de depuração e a referência de campo, consulte [Generating \(Gerando\) AWS Elemental MediaTailor registros de depuração](#)
- Para CloudWatch exemplos de consultas e análises de registros, consulte [Writing AWS Elemental MediaTailor registra diretamente no Amazon CloudWatch Logs](#)
- Para a configuração de encaminhamento de parâmetros de consulta CDN, consulte [Configure comportamentos de roteamento de CDN para MediaTailor](#)
- Para uma referência abrangente do código de erro, consulte [Solução de problemas de reprodução de MediaTailor](#)

Critérios de sucesso: quando resolvidas, as sessões devem ser inicializadas corretamente, manter uma sessão IDs consistente em todas as solicitações e os registros de depuração devem mostrar os `SESSION_INITIALIZED` eventos adequados e a geração de manifestos sem erros.

Resolva problemas de sincronização e cronometragem de pausas publicitárias da CDN para MediaTailor

AWS Elemental MediaTailor O tempo de interrupção do anúncio da rede de distribuição de conteúdo (CDN) deve ser sincronizado com precisão com os marcadores de conteúdo. Se os anúncios forem exibidos em horários incorretos ou se o horário do intervalo for inconsistente:

1. Verifique os marcadores de quebra de anúncio no conteúdo:
 - Verifique se os marcadores SCTE-35 estão colocados corretamente em seu conteúdo de origem

- Verifique se a duração do intervalo publicitário corresponde à duração real do conteúdo do anúncio
 - Confirme se o tempo de intervalo do anúncio está alinhado com os limites do conteúdo
 - Valide o formato do marcador SCTE-35 e a precisão de tempo em seus manifestos de origem
 - Teste marcadores de quebra de anúncio com diferentes tipos de conteúdo (ao vivo versus VOD)
2. Verifique o impacto do cache do CDN no tempo:
- Certifique-se de que o TTL do manifesto esteja definido como 0 para evitar desvios de tempo
 - Verifique se os parâmetros sensíveis ao tempo não estão sendo armazenados em cache
 - Verifique se há problemas de sincronização do relógio entre a fonte de conteúdo e MediaTailor a CDN
 - Monitore a variação de tempo em transmissões ao vivo de longa duração
 - Verifique a sincronização de horário do servidor de borda CDN com o NTP
3. Valide a implementação do marcador SCTE-35:
- Verifique se as EXT-X-DATERANGE etiquetas incluem as especificações corretas SCTE35 de -OUT e DURATION
 - Verifique se há marcadores SCTE35 -OUT e SCTE35 -IN emparelhados ao usar o tempo de sinalização explícito
 - Valide o alinhamento dos timestamps START-DATE com o tempo real do conteúdo
 - Teste diferentes formatos de marcadores SCTE-35 (marcadores baseados em duração versus marcadores emparelhados)
4. Teste o tempo de intervalo do anúncio em diferentes cenários:
- Compare o tempo de intervalo do anúncio com MediaTailor solicitações diretas versus solicitações de CDN
 - Teste a consistência do tempo em diferentes pontos de borda da CDN
 - Verifique o horário do intervalo publicitário com diferentes tipos de jogadores e comportamentos de buffer
 - Monitore a precisão do tempo durante os períodos de pico de tráfego
5. Depure problemas de temporização usando registros e monitoramento:
- Ative o registro de depuração para rastrear e interromper o tempo de processamento
 - Monitore CloudWatch métricas para padrões de tempo de inserção de anúncios
 - Verifique os registros da CDN para ver os padrões de solicitação relacionados ao tempo

- Use as ferramentas de depuração do player para verificar o horário do intervalo do anúncio do ponto de vista do cliente

Tolerâncias de temporização esperadas:

- O tempo de intervalo do anúncio deve estar alinhado aos marcadores SCTE-35 em seu conteúdo
- A duração do anúncio deve corresponder à duração especificada na resposta do servidor de decisão de anúncio
- A sincronização do relógio entre a fonte de conteúdo e MediaTailor a CDN deve estar dentro de 1 segundo
- O tempo do marcador SCTE-35 deve ser preciso dentro de 100 ms do tempo real do conteúdo

Recursos adicionais de solução de problemas:

- Para o formato e implementação do marcador SCTE-35, consulte [Integrando uma fonte de conteúdo para inserção de MediaTailor anúncios](#)
- Para configuração do registro de depuração e análise de tempo, consulte [Generating \(Gerando\) AWS Elemental MediaTailor registros de depuração](#)
- Para a configuração do armazenamento em cache da CDN e o impacto do tempo, consulte [Otimização de cache para CDN e integrações MediaTailor](#)
- Para procedimentos de teste abrangentes, incluindo validação de tempo, consulte [Teste e validação para CDN e integrações MediaTailor](#)
- Para monitorar o tempo e o desempenho da inserção de anúncios, consulte [Monitore as operações e o desempenho da MediaTailor CDN](#)

CrITÉRIOS de sucesso: quando resolvidos, os intervalos publicitários devem aparecer exatamente nos horários especificados pelos marcadores SCTE-35, com um tempo consistente em todos os pontos de presença da CDN e tipos de jogadores. Os registros de depuração devem mostrar um tempo preciso de processamento do ad break, sem desvios ou erros de sincronização.

Otimize o desempenho da CDN e resolva problemas de latência para MediaTailor

AWS Elemental MediaTailor o desempenho da integração da rede de distribuição de conteúdo (CDN) afeta diretamente a experiência do espectador e a qualidade da entrega de anúncios. Se você tiver tempos de resposta lentos ou degradação do desempenho:

Técnicas de medição de desempenho

Antes de solucionar problemas de desempenho, estabeleça medidas básicas e monitoramento contínuo:

1. Avalie as principais métricas de desempenho:

- Tempos de resposta: as solicitações de manifesto devem ser concluídas em 200 ms, as solicitações de segmento em 100 ms
- Taxas de acerto do cache: segmentos de conteúdo > 95%, segmentos de anúncios > 90%
- Volume de solicitações de origem: deve ser inferior a 5% do total de solicitações quando o cache é otimizado
- Tempo até o primeiro quadro: a reprodução inicial deve começar dentro de 2-3 segundos

2. Use ferramentas de medição de desempenho:

- Painéis de análise de CDN: monitore o desempenho do cache, os tempos de resposta e as taxas de erro
- CloudWatch métricas: acompanhe métricas MediaTailor de serviço, incluindo `GetManifest.Latency`
- Ferramentas para desenvolvedores de navegadores: meça o desempenho do lado do cliente e o tempo da rede
- Ferramentas de linha de comando: use curl com opções de tempo para medir solicitações específicas

3. Implemente o monitoramento contínuo:

- Configure alertas de desempenho automatizados para degradação do tempo de resposta
- Monitore o desempenho em diferentes regiões geográficas
- Monitore o desempenho durante os períodos de pico de tráfego
- Compare as métricas de desempenho antes e depois das alterações de configuração

Recursos de medição de desempenho:

- Para uma configuração abrangente de monitoramento de desempenho, consulte [Monitore as operações e o desempenho da MediaTailor CDN](#)
- Para procedimentos de teste de desempenho, consulte [Teste e validação para CDN e integrações MediaTailor](#)
- Para CloudWatch métricas e monitoramento, consulte [Monitoramento AWS Elemental MediaTailor com CloudWatch métricas da Amazon](#)

Problemas de desempenho do cache CDN

Os problemas de desempenho do cache estão entre os problemas mais comuns de integração de CDN. Esses problemas afetam todas as MediaTailor implementações e podem afetar significativamente a experiência e os custos do espectador.

Baixa taxa de acertos do cache

Sintomas: alto volume de solicitações de origem, maior latência, maiores custos de largura de banda, experiência ruim do espectador

Valores alvo:

- Segmentos de conteúdo: taxa de acerto de cache de 95% ou mais
- Segmentos de anúncios: taxa de acerto de cache de 90% ou mais
- Manifestos: varia de acordo com a implementação (manifestos personalizados não devem ser armazenados em cache)

Causas comuns:

- Configurações TTL incorretas para diferentes tipos de conteúdo
- A configuração da chave de cache inclui parâmetros de consulta desnecessários
- Cabeçalhos de controle de cache da origem não configurados corretamente
- Invalidações ou limpezas frequentes do cache
- Problemas de distribuição geográfica (conteúdo não armazenado em cache nos pontos de presença)

Soluções:

1. Revise e otimize as configurações de TTL:

- Segmentos de conteúdo: defina o TTL para corresponder à duração do segmento ou mais
- Segmentos de anúncios: defina TTL para 24 horas ou mais para anúncios reutilizáveis
- Ativos estáticos: defina TTL para 24 horas ou mais

Para recomendações abrangentes de TTL e estratégias de otimização de cache, consulte.

[Otimização de cache para CDN e integrações MediaTailor](#)

2. Otimize a configuração da chave de cache:

- Remova parâmetros de consulta desnecessários das chaves de cache
- Garanta que somente os parâmetros que afetam o conteúdo sejam incluídos
- Normalize a ordem dos parâmetros e a distinção entre maiúsculas e min

3. Verifique se os cabeçalhos de controle de cache de origem estão definidos corretamente

4. Implemente o escudo de origem (ou funcionalidade CDN equivalente) para implementações de alto tráfego. A funcionalidade Origin Shield está disponível em todos os principais CDNs , mas pode ter nomes diferentes (como CloudFront Origin Shield, Fastly Shield, Cloudflare Argo Tiered Cache). Se sua CDN não oferecer essa funcionalidade, ela poderá ser ativada MediaTailor quando você entrar em contato com o [AWS Support](#).

5. Revise as estratégias de invalidação do cache e reduza as limpezas desnecessárias

Etapas de validação:

1. Monitore as taxas de acerto do cache usando painéis de análise de CDN
2. Teste específico URLs com curl para verificar cabeçalhos de cache
3. Compare o volume da solicitação de origem antes e depois das alterações

Alto volume de solicitações de origem

Sintomas: número inesperadamente alto de solicitações chegando à MediaTailor origem, aumento da carga do servidor de origem, custos mais altos

Padrão esperado: as solicitações do Origin devem ser inferiores a 5% do total de solicitações do espectador quando as taxas de acerto do cache são ideais

Causas comuns:

- Falha no cache devido aos baixos valores de TTL
- Fragmentação da chave de cache (muitas chaves de cache exclusivas)

- Picos de tráfego geográfico em regiões sem conteúdo em cache
- Invalidações frequentes de cache

Soluções:

1. Analise os padrões de solicitação para identificar as causas da perda de cache
2. Otimize as configurações de TTL com base no tipo de conteúdo e na frequência de atualização
3. Implemente estratégias de aquecimento de cache para novos conteúdos
4. Considere a implementação do Origin Shield (disponível nas principais CDNs empresas com nomes diferentes - consulte [Implementação do Origin Shield](#) para obter detalhes)

Limite de alerta: defina alertas quando as solicitações de origem excederem 10% do total de solicitações ou aumentarem em 50% em relação à linha de base

Resolução de erros HTTP comuns

Os erros de HTTP nas integrações de CDN geralmente indicam problemas de configuração ou problemas de serviço. Esses padrões de erro são consistentes em todas as MediaTailor implementações.

404 Erros não encontrados

Sintomas: solicitações de manifesto ou segmento retornam HTTP 404, os jogadores não conseguem carregar o conteúdo, "MANIFEST_LOAD_ERROR" nos registros do player

Causas comuns:

- Configuração de origem de CDN incorreta (URL de MediaTailor endpoint incorreta)
- Padrões de caminho de comportamento de cache ausentes ou incorretos
- Problemas de reescrita de URL na configuração do CDN
- MediaTailor nome de configuração ou erros do endpoint de reprodução
- Problemas de cronometragem com conteúdo ao vivo (solicitando segmentos futuros)

Etapas de diagnóstico:

1. Teste o mesmo URL diretamente em relação à MediaTailor origem (ignore a CDN)
2. Verifique se a configuração de origem do CDN corresponde ao endpoint de MediaTailor reprodução

3. Verifique o comportamento do cache CDN, os padrões de caminho e a precedência
4. Revise os registros de acesso à CDN para obter detalhes de roteamento de solicitações
5. Validar o nome MediaTailor da configuração e as configurações da região

Soluções:

- Configuração de origem CDN correta para corresponder ao endpoint de MediaTailor reprodução
- Atualize os padrões de caminho de comportamento do cache para rotear adequadamente as solicitações
- Corrija as regras de reescrita de URL, se aplicável
- Verifique se a MediaTailor configuração existe e está ativa

403 Erros proibidos

Sintomas: solicitações retornam HTTP 403, mensagens de acesso negado, falhas de autenticação

Causas comuns:

- Parâmetros de consulta ausentes ou incorretos exigidos pelo MediaTailor
- CDN não encaminha os cabeçalhos ou parâmetros necessários
- Restrições de endereço IP ou bloqueio geográfico
- Problemas com o token de autenticação (se estiver usando assinado URLs)

Soluções:

- Verifique se todos os parâmetros de consulta necessários foram incluídos e encaminhados
- Verifique a configuração do CDN para encaminhamento de cabeçalhos e parâmetros
- Revise as restrições de IP e as configurações geográficas
- Valide tokens de autenticação e processos de assinatura

400 erros de solicitação incorreta

Sintomas: solicitações retornam HTTP 400, erros de solicitação malformada, falhas na validação de parâmetros

Causas comuns:

- Parâmetros de consulta malformados ou problemas de codificação de URL

- Valores ou formatos de parâmetros inválidos
- Faltam parâmetros necessários para MediaTailor recursos específicos
- Limitações de tamanho de URL excedidas

Soluções:

- Valide formatos de parâmetros de consulta e codificação de URL
- Verifique os valores dos parâmetros em relação aos requisitos MediaTailor da API
- Certifique-se de que todos os parâmetros necessários estejam incluídos
- Analise o tamanho do URL e considere a otimização de parâmetros

5xx Erros do servidor

Sintomas: as solicitações retornam erros HTTP 500, 502, 503 ou 504, falhas de serviço intermitentes

Causas comuns:

- MediaTailor problemas de serviço ou limitações de capacidade
- Problemas de conectividade de origem do CDN
- Problemas de tempo limite devido à lentidão nas respostas de origem
- Degradação temporária do serviço

Soluções:

- Verifique o status do AWS serviço no MediaTailor Service Health Dashboard
- Verifique as configurações de tempo limite e conectividade de origem do CDN
- Implemente a lógica de repetição com recuo exponencial
- Monitore MediaTailor CloudWatch métricas de integridade do serviço
- Entre em contato com AWS o Support se os problemas persistirem

1. Avalie o desempenho básico:

- Teste os tempos de resposta da solicitação de manifesto diretamente para MediaTailor (meta: <200 ms)
- Meça os tempos de resposta da CDN para solicitações de manifesto (meta: <100 ms para acessos ao cache)

- Verifique os tempos de carregamento do segmento tanto da origem quanto da CDN

2. Analise o desempenho da CDN:

- Verifique as taxas de acerto do cache para segmentos de conteúdo (meta: > 80% para conteúdo popular)
- Verifique se o escudo de origem (ou funcionalidade CDN equivalente) está ativado e configurado na mesma AWS região de sua origem. Diferentes CDNs usam nomes diferentes para essa funcionalidade
- Monitore o desempenho da localização periférica e a distribuição geográfica da CDN

Benchmarks de desempenho:

- Monitore os tempos de resposta da geração de manifestos e compare com seu desempenho básico
- Os acessos ao cache do CDN são significativamente mais rápidos do que as solicitações de origem
- Os tempos de resposta do ADS não devem causar atrasos na geração de manifestos

Recursos adicionais de solução de problemas:

- Para estratégias abrangentes de otimização de desempenho, consulte [Guia de otimização de desempenho para CDN e integrações MediaTailor](#)
- Para obter detalhes sobre a implementação do Origin Shield, consulte [Implementação do Origin Shield](#)
- Para otimizar o cache do CDN, consulte [Otimização de cache para CDN e integrações MediaTailor](#)
- Para monitoramento e métricas de desempenho, consulte [Monitore as operações e o desempenho da MediaTailor CDN](#)
- Para procedimentos de teste de desempenho, consulte [Teste e validação para CDN e integrações MediaTailor](#)

Critérios de sucesso: quando resolvidos, os tempos de resposta devem atingir os benchmarks desejados (manifestos menores que 200 ms, segmentos menores que 100 ms), as taxas de acerto do cache devem exceder 90% para a maioria dos tipos de conteúdo e o volume de solicitações de origem deve ser inferior a 5% do total de solicitações. O desempenho deve ser consistente em todas as regiões geográficas e tipos de dispositivos.

Corrija o comportamento inconsistente da CDN em todos os dispositivos e plataformas para MediaTailor

AWS Elemental MediaTailor a integração da rede de distribuição de conteúdo (CDN) deve fornecer uma entrega consistente de anúncios em todos os dispositivos e plataformas. Se os anúncios se comportarem de forma diferente em todos os dispositivos:

1. Garanta o encaminhamento consistente de cabeçalhos em todos os comportamentos de CDN.
 - Verifique se os cabeçalhos User-Agent e de segmentação personalizada são encaminhados de forma consistente X-Forwarded-For
 - Verifique se as regras de encaminhamento de cabeçalho se aplicam a todos os comportamentos de cache relevantes
2. Verifique a compatibilidade do player com sua configuração de CDN.
 - Teste com vários tipos de jogadores (HLS.js, Video.js, jogadores nativos) para identificar problemas específicos do jogador
 - Verifique os requisitos de cabeçalho específicos do jogador ou as diferenças no tratamento de URL
3. Teste com vários tipos de dispositivos para identificar problemas específicos do dispositivo.
 - Inclua dispositivos móveis, tablets, navegadores inteligentes TVs e de desktop em seus testes
 - Teste diferentes sistemas operacionais e versões de navegadores
 - Verifique se a segmentação de anúncios específica do dispositivo funciona corretamente

Se você seguiu essas etapas de solução de problemas e ainda precisa de ajuda, consulte [Obtenha suporte para integração de CDN](#).

Preparação para solução de

Configure ferramentas e processos para simplificar a solução de problemas quando surgirem problemas de integração com a CDN. A preparação proativa torna a solução de problemas mais rápida e eficaz quando ocorrem problemas.

Permita um registro abrangente

Registros detalhados são essenciais para diagnosticar problemas de integração de CDN. Configure o registro para capturar as informações necessárias durante a solução de problemas.

1. Habilite registros detalhados de acesso à CDN:

- Configure o registro em log para todos os comportamentos de cache que lidam com MediaTailor solicitações
- Inclua cadeias de caracteres de consulta e cabeçalhos personalizados nas entradas de registro
- Configure ferramentas de análise de registros para identificar padrões e anomalias
- Ative registros em tempo real para detecção imediata de problemas durante eventos ao vivo
- Configure políticas de retenção de registros para manter dados históricos de solução de problemas

2. Configurar MediaTailor o registro:

- Ative os registros de acesso para suas MediaTailor configurações
- Configure grupos de CloudWatch registros para gerenciamento centralizado de registros
- Configure filtros de log para identificar padrões de erro

3. Configure o registro do servidor de origem:

- Habilite registros de acesso detalhados em seus servidores de origem de conteúdo
- Inclua cabeçalhos de solicitação e códigos de resposta nos registros
- Monitore as métricas de desempenho do servidor de origem

Adicionar cabeçalhos de solicitação de diagnóstico

Cabeçalhos personalizados ajudam a rastrear solicitações por meio de sua CDN e identificar problemas de roteamento.

1. Configure os cabeçalhos de diagnóstico da CDN:

- Adicione um identificador exclusivo a cada solicitação (por exemplo, X-Request-ID)
- Inclua informações específicas da CDN nos cabeçalhos da solicitação
- Adicione informações de localização de borda ou POP (ponto de presença) para rastrear o roteamento geográfico
- Inclua cabeçalhos de status do cache (Hit, Miss, RefreshHit) para análise do comportamento do cache

2. Adicione cabeçalhos de resposta para depuração:

- Incluir cabeçalhos de identificação do servidor
- Adicione informações de tempo para análise de desempenho
- Incluir cabeçalhos de controle de cache para solicitações de manifesto

Estabeleça métricas básicas de desempenho

Documente as faixas normais de desempenho para identificar rapidamente anomalias durante a solução de problemas:

1. Registre as métricas básicas:

- Taxas de acerto do cache para diferentes tipos de conteúdo
- Percentis de tempo de resposta (P50, P95, P99)
- Taxas de erro por código de status
- Solicite padrões de volume por hora do dia

2. Documente as expectativas de desempenho:

- Alvo taxas de acerto do cache (mais de 95% para conteúdo, mais de 90% para anúncios)
- Tempos de resposta aceitáveis (<100 ms em cache, <500 ms de origem)
- Taxas de erro máximas aceitáveis (< 1% para 4xx, < 0,1% para 5xx)

3. Crie painéis de desempenho: configure painéis de monitoramento que mostrem as métricas atuais em comparação com os valores básicos.

Prepare ferramentas de solução de problemas

Configure as ferramentas e as permissões de acesso necessárias para uma solução eficaz de problemas:

1. Ferramentas de linha de comando:

- `curl` para testar solicitações e respostas HTTP
- `dig` ou `nslookup` para solução de problemas de DNS
- Ferramentas de validação HLS/DASH para verificação de manifestos
- Ferramentas de análise de registros (`grep`, `awk` ou analisadores de registros especializados)

2. Permissões de acesso:

- Acesso ao console de gerenciamento CDN para análise de configuração
- MediaTailor acesso ao console para verificação de configuração
- CloudWatch acesso para métricas e análise de registros
- Acesso ao servidor de origem para solução de problemas de back-end

- Diagramas de arquitetura de rede
- CDN e documentação MediaTailor de configuração
- Informações de contato para procedimentos de escalonamento
- Runbooks de solução de problemas para cenários comuns

Guias de solução de problemas específicos do fluxo de trabalho

Este guia universal de solução de problemas aborda problemas comuns em todas as integrações de MediaTailor CDN. Para problemas específicos de fluxos de trabalho ou serviços específicos, consulte estes recursos especializados de solução de problemas:

Solução de problemas de inserção de anúncios no lado do servidor (SSAI)

Para problemas específicos de SSAI, incluindo falhas de inserção de anúncios, problemas de transição de anúncios e problemas de personalização, consulte a documentação de solução de problemas de SSAI específica do fluxo de trabalho.

Problemas comuns específicos do SSAI:

- Falhas na inserção de anúncios e intervalos de anúncios vazios
- Problemas de sincronização e tempo de transição de anúncios
- Problemas de personalização e segmentação
- Discrepâncias de rastreamento e análise de anúncios

Solução de problemas de montagem de canais

Para problemas específicos de montagem de canais, incluindo problemas de geração de manifestos e funcionalidade de mudança de horário, consulte a documentação do fluxo de trabalho de montagem de canais.

Problemas comuns de montagem de canais:

- Erros de geração e compilação de manifestos
- Janela de mudança de horário e problemas de funcionalidade do DVR
- Disponibilidade do conteúdo de origem e problemas de failover
- Programação do programa e sincronização de metadados

MediaPackage solução de problemas de integração

Para problemas MediaPackage específicos, incluindo filtragem de manifestos e problemas de endpoint EMP, consulte [Solução de problemas de integração de CDN](#)

Problemas comuns de MediaPackage integração:

- Erros de parâmetros de filtragem de manifestos
- MediaPackage problemas de conectividade de endpoint
- Problemas de comportamento de cache específicos do EMP
- MediaPackage problemas de autenticação de origem

CloudFront solução de problemas específica

Para problemas CloudFront específicos de configuração e configuração, consulte a documentação de CloudFront integração.

CloudFront Problemas comuns:

- Configuração de distribuição e configuração do comportamento do cache
- Identidade de acesso e configuração de segurança do Origin
- CloudFront códigos de erro e respostas específicos
- Restrições geográficas e problemas de localização periférica

Recursos adicionais:

- Para obter orientação sobre otimização de desempenho, consulte [Otimização do desempenho da CDN](#)
- Para configuração de monitoramento e alertas, consulte [Monitoramento de CDN](#)
- Para obter suporte e assistência gerais, consulte [Obtenha suporte para integração de CDN](#)

Análise de log de integração de CDN e referência de código de erro para MediaTailor

AWS Elemental MediaTailor os registros de integração da rede de distribuição de conteúdo (CDN) fornecem informações valiosas sobre desempenho e erros. Este guia abrange tanto os registros da CDN (da sua rede de distribuição de conteúdo) quanto MediaTailor os registros e códigos de erro que são relevantes para a solução de problemas de integração da CDN. Use essa referência quando

precisar entender o que os registros e códigos de erro da rede de distribuição de conteúdo estão dizendo sobre sua MediaTailor integração. Este guia ajuda você a interpretar entradas de registro e mensagens de erro para identificar a causa raiz dos problemas.

Tópicos relacionados:

- Para procedimentos step-by-step de solução de problemas, consulte [Solucionar problemas de integração com a CDN](#)
- Para estratégias proativas de monitoramento e prevenção, consulte [Monitore as operações e o desempenho da MediaTailor CDN](#)
- Para escalar e obter ajuda adicional, consulte [Obtenha suporte para integração de CDN](#)
- Para obter recursos e configurações de MediaTailor registro abrangentes, consulte [MediaTailor recursos de configuração de log](#)
- Para referência ao formato de CloudFront registro, consulte [formato de registro de CloudFront acesso](#)
- Para referência do código de status HTTP, consulte [Códigos de status de resposta HTTP](#)

Técnicas de interpretação e análise de registros CDN para MediaTailor

AWS Elemental MediaTailor a integração da rede de entrega de conteúdo (CDN) gera registros detalhados que ajudam você a entender os padrões de solicitação e identificar problemas. Ao analisar os registros da rede de distribuição de conteúdo, concentre-se nesses indicadores-chave:

Observação: os exemplos a seguir usam a CloudFront terminologia da Amazon. Outros provedores de CDN podem usar termos diferentes para conceitos semelhantes.

Códigos de status de HTTP

- 200- Solicitação bem-sucedida
- 404- Recurso não encontrado (verifique as regras e a MediaTailor configuração de roteamento)
- 403- Acesso negado (verifique as configurações de segurança do CDN e as permissões de origem)
- 502/503/504- Erros do servidor de origem (verifique a integridade do MediaTailor serviço e a conectividade de origem)

Indicadores de comportamento do cache

- Hit- Conteúdo servido a partir do cache CDN

- **Miss**- Conteúdo obtido da origem
- **RefreshHit**- Conteúdo em cache validado com origem
- **Error**- A solicitação resultou em um erro

Solicite padrões para monitorar

- As solicitações de manifesto normalmente devem resultar em Miss ou RefreshHit devido a configurações de TTL baixas
- Segmentos de conteúdo devem ser exibidos Hit para conteúdo popular
- Segmentos de anúncios podem ser exibidos Miss devido à personalização

Monitoramento da integridade do serviço

Ao analisar os registros da CDN e solucionar problemas, verifique a integridade do MediaTailor serviço para determinar se os problemas estão relacionados ao serviço:

AWS Service Health Dashboard

Verifique o status atual do MediaTailor serviço e quaisquer eventos de serviço em andamento

Acesso: [AWS Service Health Dashboard](#)

Use isso quando: Você vê erros 5xx generalizados ou tempos limite de serviço nos registros do CDN

AWS Personal Health Dashboard

Visualize notificações de integridade de serviços e eventos de manutenção específicos da conta

Acesso: [AWS Personal Health Dashboard](#)

Use isso quando: Você precisar de informações de integridade do serviço ou notificações de manutenção específicas da conta

Monitoramento de CloudWatch métricas da Amazon

Monitore os indicadores de integridade do MediaTailor serviço por meio de CloudWatch métricas:

- `GetManifest.Errors`- Rastreie erros de geração de manifestos
- `GetManifest.Latency`- Monitore o desempenho do tempo de resposta
- `AdDecisionServer.Errors`- Monitore problemas de conectividade do servidor de anúncios

- `Origin.Errors`- Rastreie problemas de conectividade do servidor de origem

Acesso: [CloudWatchConsole](#)

Use isso quando: Você precisar de métricas detalhadas de desempenho do serviço e tendências históricas

Fluxo de trabalho de solução de problemas de integridade do

1. Verifique o AWS Service Health Dashboard para ver o status atual do serviço
2. Consulte o AWS Personal Health Dashboard para obter notificações específicas da conta
3. Monitore CloudWatch métricas para indicadores de desempenho do serviço
4. Correlacione o status de integridade do serviço com padrões de log e taxas de erro da CDN
5. Se a integridade do serviço estiver normal, concentre-se na configuração da CDN e nos problemas do servidor de origem

Recursos de análise de registros do CDN

- [CloudFront registros de acesso](#) - Guia completo para campos e formatos de log CDN
- [CloudFront registros em tempo real](#) - Streaming e análise de registros em tempo real
- [Análise de dados de registro com o CloudWatch Logs Insights](#) - Consulta e análise avançadas de registros
- [Monitore os recursos da carga de trabalho](#) - Padrões de monitoramento Well-Architected

Tópicos

- [MediaTailor códigos de erro e solução de problemas de integração de CDN](#)
- [Ferramentas de análise de registros CDN e técnicas de monitoramento para MediaTailor](#)
- [MediaTailor recursos de configuração de log](#)

MediaTailor códigos de erro e solução de problemas de integração de CDN

AWS Elemental MediaTailor os códigos de erro fornecem informações específicas sobre problemas de integração quando transmitidos pela sua rede de distribuição de conteúdo (CDN). Quando MediaTailor retorna erros por meio de sua rede de distribuição de conteúdo, esses códigos indicam problemas específicos:

Observação: esses códigos de erro são retornados MediaTailor e passados por sua CDN. Para erros específicos da CDN, consulte a documentação do seu provedor de CDN.

400 solicitação inválida

Causas comuns: parâmetros de consulta malformados, falta de parâmetros obrigatórios, sessão inválida IDs

Verifique: configuração de encaminhamento de string de consulta CDN, codificação de parâmetros

403 proibido

Causas comuns: o player cliente está solicitando um segmento que não existe ou não tem permissão para acessar. Para segmentos de anúncios, MediaTailor pode ter especificado um segmento inexistente (entre em contato com o AWS [Support](#) para investigar). Para segmentos de conteúdo, restrições de acesso do provedor de origem ou problemas de autenticação

Verifique: Para segmentos de anúncios: entre em contato com o [AWS Support](#) se MediaTailor estiver gerando um segmento URLs inválido. Para segmentos de conteúdo: verifique as permissões do servidor de origem, as credenciais de autenticação e as políticas de acesso. Verifique se os segmentos existem no local solicitado URLs

404 Not Found (404 Não encontrado)

Causas comuns: a MediaTailor configuração não existe ou está inativa, caminho de URL de reprodução incorreto, solicitações de manifesto ou segmento para recursos inexistentes

Verifique: Verifique se a MediaTailor configuração existe e está ativa, confirme se o URL de reprodução corresponde ao formulário `GetPlaybackConfiguration`, verifique se as regras `ManifestEndpointPrefix` de roteamento da CDN encaminham as solicitações para os endpoints corretos MediaTailor

500 Internal Server Error

Causas comuns: problemas no servidor de origem, problemas de conectividade do ADS, erros de processamento do manifesto

Verifique: integridade do servidor de origem, validade da resposta do ADS, status do MediaTailor serviço

502 Gateway inválido

Causas comuns: servidor de origem inacessível, resposta de origem inválida

UnsupportedManifestException devido a problemas de alinhamento da playlist HLS (marcadores SCTE não alinhados nas playlists, falta de marcadores SCTE em algumas playlists, tempo de interrupção de anúncios inconsistente nas variantes de taxa de bits)

Verifique: regras de firewall e conectividade do servidor Origin, resolução de DNS, consistência da playlist HLS em todas as variantes de taxa de bits, alinhamento do marcador SCTE-35 nos mesmos segmentos em todas as playlists, verifique se todas as playlists contêm os marcadores de quebra de anúncio necessários

Recursos de análise de código de erro:

- [Solução de problemas MediaTailor](#) - Referência completa MediaTailor do código de erro
- [Solução de problemas CloudFront de respostas de erro](#) - análise de erros específica da CDN
- [Códigos de status de resposta HTTP](#) - Referência abrangente do código de status HTTP
- [Gerenciamento de falhas](#) - Padrões de tratamento de erros do Well-Architected

Ferramentas de análise de registros CDN e técnicas de monitoramento para MediaTailor

AWS Elemental MediaTailor a integração da rede de distribuição de conteúdo (CDN) gera grandes volumes de dados de log que exigem ferramentas e técnicas de análise eficientes. Use essas abordagens para analisar com eficiência a rede e os MediaTailor registros de entrega de conteúdo:

- Análise de linha de comando: use ferramentas como `grep`, `awk`, e `sort` para filtrar e analisar padrões de log
- Amazon CloudWatch Logs Insights: consulte CDN e MediaTailor registros com sintaxe semelhante a SQL para análise avançada
- Ferramentas de terceiros: considere plataformas de análise de registros para monitoramento e alertas abrangentes
- Painéis personalizados: crie visualizações que combinam métricas de CDN com dados de desempenho MediaTailor

Se precisar de ajuda adicional com análise de registros ou interpretação de padrões de erro complexos, consulte [Obtenha suporte para integração de CDN](#).

Ferramentas e recursos de análise de registros:

- [CloudWatch Logs Insights](#) - consultas semelhantes a SQL para análise de registros
- [Amazon OpenSearch Service](#) - Pesquisa e análise avançadas de registros
- [CloudWatch painéis](#) - Visualização e monitoramento personalizados
- [Instrumentando sistemas distribuídos para visibilidade operacional](#) - Padrões avançados de observabilidade
- Observabilidade da [carga de trabalho do projeto - Guia de observabilidade](#) da Well-Architected

MediaTailor recursos de configuração de log

Além dos registros de CDN, MediaTailor fornece recursos de registro abrangentes para monitorar inserção de anúncios, geração de manifestos e interações de serviço. Use esses recursos para configurar e analisar MediaTailor registros:

MediaTailor registros de manifesto e serviço

Configure e analise registros para geração de manifestos, interações de origem e eventos de serviço:

- [AWS Elemental MediaTailor descrição dos registros de manifesto e tipos de eventos](#)- Referência completa para registros de MediaTailor manifestos e tipos de eventos
- [Visualizar AWS Elemental MediaTailor logs](#)- Guia para visualizar e interpretar registros MediaTailor

Configuração de registros vendidos

Configure a entrega flexível de registros para vários destinos com otimização de custos:

- [Usando registros vendidos para enviar AWS Elemental MediaTailor logs](#)- Configurar a entrega de registros para Amazon S3, Firehose ou Logs CloudWatch
- [Migrando seu AWS Elemental MediaTailor estratégia de registro](#)- Guia de migração para configurações de registro existentes

CloudWatch Integração de registros

Integre MediaTailor os registros CloudWatch para monitoramento e análise:

- [Visualizar AWS Elemental MediaTailor logs](#)- Configuração e análise de CloudWatch registros
- [CloudWatchInsights](#)- Consulta e análise avançadas de registros

Registro específico de anúncios

Monitore o desempenho da inserção de anúncios e solucione problemas relacionados a anúncios:

- Registros de interação do ADS - Monitore a comunicação e os erros do servidor de decisões de anúncios

Principais diferenças entre CDN e MediaTailor registros:

- Registros de CDN: mostre request/response padrões, comportamento de cache e erros em nível de rede da sua rede de distribuição de conteúdo
- MediaTailor registros: mostre detalhes de inserção de anúncios, eventos de geração de manifestos, interações de origem e erros específicos do serviço
- Análise combinada: use os dois tipos de log juntos para obter visibilidade completa do seu fluxo de trabalho de streaming

Para um monitoramento abrangente que combina CDN e MediaTailor registro, consulte [Monitore as operações e o desempenho da MediaTailor CDN](#).

Automatize MediaTailor e CDN com CloudFormation

A automatização AWS Elemental MediaTailor com uma rede de distribuição de conteúdo (CDN) AWS CloudFormation simplifica seu fluxo de trabalho de inserção de anúncios e melhora a confiabilidade.

Esta seção mostra como usar AWS CloudFormation (AWS infraestrutura como serviço de código) para configurar automaticamente uma rede AWS Elemental MediaTailor de distribuição de conteúdo (CDN). Embora você possa configurar manualmente essa integração conforme descrito nas seções anteriores, o uso CloudFormation economiza tempo e reduz erros ao automatizar todo o processo com um único modelo.

Se você é novato CloudFormation, é um serviço que permite criar um arquivo de modelo que define todos os recursos da AWS de que você precisa. Quando você implanta esse modelo, cria e configura CloudFormation automaticamente esses recursos para você, garantindo que eles funcionem juntos corretamente.

Para obter mais informações sobre CloudFormation, consulte o [Guia CloudFormation do usuário](#). Para obter informações sobre os tipos de MediaTailor recursos em CloudFormation, consulte [AWS: Referência do tipo de MediaTailor recurso](#).

Tópicos

- [Por que usar CloudFormation para MediaTailor integração com CDN](#)
- [Prepare-se para a CloudFormation implantação de CDN e MediaTailor integrações](#)
- [Implante o CloudFormation modelo para CDN e integrações MediaTailor](#)
- [Use os recursos CloudFormation implantados para CDN e integração MediaTailor](#)
- [Teste e valide sua CloudFormation implantação para CDN e integração MediaTailor](#)
- [Solucione problemas comuns CloudFormation de implantação para CDN e integrações MediaTailor](#)
- [Personalize o CloudFormation modelo para CDN e integrações MediaTailor](#)
- [AWS CloudFormation referência de modelo AWS Elemental MediaTailor e CloudFront integração com a Amazon](#)

Por que usar CloudFormation para MediaTailor integração com CDN

AWS Elemental MediaTailor a automação AWS CloudFormation oferece vantagens significativas para profissionais de transmissão que gerenciam fluxos de trabalho de streaming. A configuração manual MediaTailor com uma rede de distribuição de conteúdo (CDN) pode ser demorada e propensa a erros. O uso da CloudFormation automação oferece as seguintes vantagens.

- **Consistência:** garante que a mesma configuração seja sempre implantada, reduzindo o erro humano.
- **Controle de versão:** armazene sua infraestrutura como código em um sistema de controle de versão para rastrear alterações.
- **Implantação rápida:** implante configurações complexas em minutos, em vez de horas de configuração manual.
- **Replicação do ambiente:** replique facilmente as configurações em ambientes de desenvolvimento, teste e produção.
- **Documentação:** O modelo em si serve como documentação da sua infraestrutura.

Veja como o fluxo de trabalho automatizado se compara à configuração manual:

Configuração manual (várias etapas)	Configuração automatizada (modelo único)
Criar MediaTailor configuração de reprodução	Implemente um CloudFormation modelo com parâmetros
Crie uma CloudFront distribuição	
Configurar comportamentos de cache	
Definir configurações de segurança	

O fluxo de trabalho automatizado para configuração MediaTailor CloudFront segue estas etapas:

1. Implante o CloudFormation modelo com sua origem de conteúdo e parâmetros do servidor de anúncios
2. CloudFormation cria e configura todos os recursos necessários:
 - MediaTailor configuração de reprodução para inserção de anúncios
 - CloudFront distribuição com comportamentos de cache apropriados
 - Configurações de segurança para proteção de conteúdo
3. Use as CloudFormation saídas para acessar seu stream habilitado para anúncios URLs
4. Transmita seu conteúdo com anúncios inseridos dinamicamente

Prepare-se para a CloudFormation implantação de CDN e MediaTailor integrações

AWS Elemental MediaTailor a implantação com AWS CloudFormation requer pré-requisitos específicos e etapas de preparação. Antes de começar a trabalhar com CloudFormation a integração MediaTailor com a Amazon CloudFront, verifique se você tem o seguinte.

- Uma AWS conta com permissões para criar MediaTailor e CloudFormation recursos CloudFront
- Uma origem de conteúdo na qual seu conteúdo de vídeo é hospedado (como AWS Elemental MediaPackage Amazon S3 ou outro servidor de origem)
- Um servidor de decisão de anúncios (ADS) que pode responder a solicitações VAST

Antes de implantar o CloudFormation modelo, reúna estes parâmetros necessários:

AdServerUrl

URL do servidor de anúncios VAST para inserção dinâmica de anúncios. Um endpoint VAST estático é fornecido para teste.

ContentOriginDomainName

Nome de domínio da origem do seu conteúdo sem protocolo (por exemplo *mediapackage-domain.mediapackagev2.us-west-2.amazonaws.com*, *mybucket.s3.amazonaws.com*, ou *custom-origin.example.com*). Não inclua prefixos `http://` ou `https://` nem nenhum caminho.

ContentOriginType

O tipo de origem do conteúdo:

- `mediapackagev2`: Para origens AWS Elemental MediaPackage
- `s3`: Para origens de buckets do Amazon S3
- `personalizado`: Para qualquer outro tipo de origem

O modelo criará vários AWS recursos que funcionam juntos para fornecer seu conteúdo com anúncios personalizados. A seguir, descrevemos o que cada componente faz:

Controle de acesso do Origin

O Origin Access Control (OAC) é um recurso de segurança que garante que seu conteúdo só possa ser acessado por meio do servidor de origem CloudFront, e não diretamente dele. Isso ajuda a proteger seu conteúdo contra acesso não autorizado.

Para MediaPackage origens do Amazon S3, o modelo cria um recurso Origin Access Control (OAC) para proteger o acesso ao seu conteúdo.

MediaTailor configuração de reprodução

A configuração de MediaTailor reprodução é o componente principal que manipula a inserção de anúncios. Ele recebe conteúdo de sua origem, solicita anúncios do seu servidor de anúncios e os combina em um fluxo personalizado para cada espectador.

O modelo cria uma configuração de MediaTailor reprodução com as seguintes configurações:

- Fonte de conteúdo de vídeo apontando para sua CloudFront distribuição

- URL do servidor de decisão de anúncio configurado para seu endpoint VAST especificado
- Configuração de pré-lançamento ao vivo para inserção de anúncios durante transmissões ao vivo
- Configuração de CDN com prefixos de URL de segmento apropriados

CloudFront distribuição

A CloudFront distribuição entrega seu conteúdo para espectadores em todo o mundo com baixa latência. Ele lida com diferentes tipos de solicitações (manifestos, segmentos de conteúdo, segmentos de anúncios) e as encaminha para as origens apropriadas.

Para profissionais de radiodifusão CDNs iniciantes, aqui estão alguns termos-chave:

Origem

Um servidor onde seu conteúdo original é armazenado (como MediaPackage ou Amazon S3)

Comportamento de cache

Regras que determinam como diferentes tipos de conteúdo são armazenados em cache e entregues

Política de cache

Configurações que controlam por quanto tempo o conteúdo é armazenado em cache e quais componentes da solicitação afetam o armazenamento em cache

O modelo cria uma CloudFront distribuição com os seguintes componentes:

- Três origens:
 - Origem do conteúdo (MediaPackage Amazon S3 ou personalizado)
 - MediaTailor manifesta origem
 - MediaTailor origem dos segmentos
- Comportamentos de cache com padrões apropriados:
 - Comportamento padrão para segmentos de conteúdo
 - Comportamento para segmentos de MediaTailor anúncios (/tm/*)
 - Comportamento para mídia MediaTailor intersticial (/v1/i-media/*)
 - Comportamento para manifestos personalizados (/v1/*)
 - Comportamento das solicitações de redirecionamento de segmentos (/segment/*)

- Políticas de cache otimizadas para cada comportamento:
 - `CachingOptimized` para conteúdo armazenável em cache
 - `CachingDisabled` para manifestos personalizados
- Políticas de solicitação de origem para garantir o encaminhamento adequado do cabeçalho
- Políticas de cabeçalho de resposta para suporte ao CORS

Implante o CloudFormation modelo para CDN e integrações MediaTailor

AWS Elemental MediaTailor a implantação usando o AWS CloudFormation modelo é simples quando você entende o que o modelo criará. Esse processo leva cerca de 15 a 30 minutos, com a maior parte do tempo aguardando a implantação da CloudFront distribuição.

Para implantar o CloudFormation modelo e configurar seu fluxo de trabalho automatizado de inserção de anúncios:

Para implantar o MediaTailor CloudFormation modelo

1. Baixe o CloudFormation modelo do AWS Elemental MediaTailor GitHub repositório ou copie-o do [AWS CloudFormation referência de modelo AWS Elemental MediaTailor e CloudFront integração com a Amazon](#).
2. Abra o [console do CloudFormation](#).
3. Escolha Criar pilha > Com novos recursos (padrão).
4. Em Especificar modelo, escolha Carregar um arquivo de modelo e fazer upload do modelo.
5. Insira o nome da pilha e forneça valores para os parâmetros necessários:
 - `AdServerUrl`: URL do seu servidor de anúncios VAST (por exemplo, `https://your-ad-server.com/vast`)
 - `ContentOriginDomainName`: nome de domínio da origem do seu conteúdo sem protocolo (por exemplo, `mediapackage-domain.mediapackagev2.us-west-2.amazonaws.com`)
 - `ContentOriginType`: Selecione o tipo de origem do conteúdo:
 - `mediapackagev2`: Para origens AWS Elemental MediaPackage
 - `s3`: Para origens de buckets do Amazon S3
 - `personalizado`: Para qualquer outro tipo de origem
6. Revise a configuração e escolha Criar pilha.

7. Aguarde a conclusão da criação da pilha, o que normalmente leva de 5 a 10 minutos. Você pode monitorar o progresso na guia Eventos.
8. Depois de concluído, navegue até a guia Saídas para encontrar os manifestos URLs HLS e DASH.

 Note

Se você estiver usando AWS Elemental MediaPackage como origem de conteúdo, certifique-se de que seus MediaPackage endpoints estejam configurados e acessíveis adequadamente. Para obter mais informações, consulte [MediaPackage Integração CDN](#).

Use os recursos CloudFormation implantados para CDN e integração MediaTailor

AWS Elemental MediaTailor os recursos implantados pela AWS CloudFormation pilha fornecem vários resultados importantes que você usará para acessar seu conteúdo com a inserção de anúncios. Depois que a CloudFormation pilha for criada com sucesso, você precisará entender como usar as saídas para acessar seu conteúdo com anúncios inseridos. Isso é semelhante ao que você usaria MediaTailor URLs em uma configuração manual, mas a CloudFormation implantação os fornece URLs automaticamente.

Após a implantação bem-sucedida, a CloudFormation pilha fornece vários resultados importantes que você usará para acessar seu conteúdo com a inserção de anúncios:

CloudFrontDomainName

O nome de domínio da sua CloudFront distribuição (por exemplo, *d1234abcdef.CloudFront.net*)

HlsManifestUrl

URL base para manifestos HLS com inserção de anúncios (por exemplo, *https://d1234abcdef.CloudFront.net12345/v1/master/// my-playback-config*)

DashManifestUrl

URL base para manifestos DASH com inserção de anúncios (por exemplo, *https://d1234abcdef.CloudFront.net12345/v1/dash/// my-playback-config*)

MediaTailorPlaybackConfigName

Nome da configuração de MediaTailor reprodução criada (como *my-stack-PlaybackConfig*)

Construa a reprodução URLs

Para criar o URL de reprodução completo do seu conteúdo com anúncios, você precisará combinar o URL base das CloudFormation saídas com seu caminho de manifesto específico. Essa é uma etapa fundamental para os profissionais de radiodifusão entenderem, pois ela conecta seu conteúdo existente ao sistema de inserção de anúncios.

1. Comece com o URL de manifesto apropriado a partir das saídas:

```
HlsManifestUrl: https://d1234abcdef.CloudFront.net/v1/master/12345/my-playback-config/
```

2. Anexe seu caminho de manifesto específico:

```
Your manifest path: channel/index.m3u8
```

3. O URL de reprodução completo se torna:

```
https://d1234abcdef.CloudFront.net/v1/master/12345/my-playback-config/channel/index.m3u8
```

Use esse URL no seu player de vídeo para reproduzir conteúdo com anúncios inseridos dinamicamente.

Tip

Se você não tiver certeza de qual deve ser o caminho do manifesto, verifique seu servidor de origem. Para MediaPackage origens, esse é o caminho para o manifesto HLS ou DASH do seu endpoint. Para origens do Amazon S3, esse é o caminho para seu arquivo de manifesto dentro do bucket.

Para obter mais informações sobre a estrutura de MediaTailor URL, consulte [Configure a integração da CDN com MediaTailor](#).

Configurar um player de vídeo

Depois de ter seu URL de reprodução, você precisa configurar um player de vídeo para usá-lo. Para profissionais de transmissão, isso é semelhante à configuração de um player para qualquer stream HLS ou DASH, mas agora o stream incluirá anúncios personalizados. Aqui está um exemplo simples usando o popular HLS.js player:

```
<!DOCTYPE html>
<html>
<head>
  <title>MediaTailor Playback Example</title>
  <script src="https://cdn.jsdelivr.net/npm/hls.js@latest"></script>
</head>
<body>
  <video id="video" controls style="width: 640px; height: 360px;"></video>

  <script>
    const video = document.getElementById('video');
    const mediaUrl = 'https://<replaceable>d1234abcdef.CloudFront.net</replaceable>/v1/master/<replaceable>12345</replaceable>/<replaceable>my-playback-config</replaceable>/<replaceable>channel/index.m3u8</replaceable>';

    if (Hls.isSupported()) {
      const hls = new Hls();
      hls.loadSource(mediaUrl);
      hls.attachMedia(video);
    } else if (video.canPlayType('application/vnd.apple.mpegurl')) {
      video.src = mediaUrl;
    }
  </script>
</body>
</html>
```

Você também pode usar reprodutores de transmissão profissionais como:

- Jogador JW
- Jogador Bitmovin
- THEOplayer
- Video.js

Para obter mais informações sobre a integração do player com MediaTailor, consulte [MediaTailor requisitos de integração do servidor de anúncios](#).

Teste e valide sua CloudFormation implantação para CDN e integração MediaTailor

AWS Elemental MediaTailor a validação da implantação é uma etapa crítica para os profissionais de radiodifusão antes da transmissão ao vivo. Esta seção orienta você a testar sua implantação para garantir que os anúncios sejam inseridos corretamente e que o conteúdo seja entregue sem problemas.

Depois de implantar o CloudFormation modelo, siga estas etapas para verificar se sua configuração está funcionando corretamente:

Para testar sua MediaTailor CloudFront integração

1. Verifique se todos os recursos foram criados com êxito no CloudFormation console.
2. Verifique se a configuração de MediaTailor reprodução está ativa no [MediaTailor console](#).
3. Verifique se a CloudFront distribuição está implantada e habilitada no [CloudFront console](#).
4. Teste a reprodução usando um exemplo de manifesto:
 1. Crie o URL de reprodução completo conforme descrito em [Construa a reprodução URLs](#).
 2. Use um player de vídeo compatível com HLS ou DASH (como VLC, JW Player ou console player). AWS
 3. Verifique se o conteúdo é reproduzido e os anúncios são inseridos nos pontos de interrupção esperados.
5. Verifique se há erros de inserção de anúncios nos MediaTailor CloudWatch logs.

Ao testar a inserção de anúncios, procure esses indicadores de sucesso:

- Transições suaves entre conteúdo e anúncios
- Os anúncios aparecem nos pontos de interrupção esperados (pré-lançamento, meio lançamento, pós-lançamento)
- A qualidade do anúncio corresponde à qualidade do conteúdo
- Sem erros de buffer ou reprodução durante as transições de anúncios

Para obter procedimentos de teste mais detalhados, consulte [Compreender o AWS Elemental MediaTailor comportamento de inserção de anúncios](#). Para testes e validação abrangentes da integração de CDN, consulte [Teste e validação para CDN e integrações MediaTailor](#).

Solucione problemas comuns CloudFormation de implantação para CDN e integrações MediaTailor

AWS Elemental MediaTailor problemas de implantação podem ocorrer mesmo com a automação durante a implantação ou a reprodução. Como profissional de radiodifusão, entender como solucionar esses problemas ajudará você a manter um serviço de streaming confiável com inserção de anúncios.

Se você encontrar problemas com sua CloudFormation implantação ou com o resultado MediaTailor e a CloudFront integração, consulte estes problemas e soluções comuns:

CloudFormation problemas de implantação

A criação da pilha falha com o erro “Falha na criação do recurso”

Causas possíveis:

- Formato de nome de domínio de origem de conteúdo inválido
- Permissões insuficientes para criar recursos

Solução: verifique o erro específico do recurso na guia de CloudFormation eventos. Verifique se o nome do domínio de origem do conteúdo está formatado corretamente sem prefixos ou caminhos de protocolo. Certifique-se de que sua função do IAM tenha permissões suficientes para criar todos os recursos necessários.

CloudFront a distribuição leva muito tempo para ser implantada

Causa: CloudFront as distribuições normalmente levam de 15 a 30 minutos para serem totalmente implantadas.

Solução: Este é o comportamento normal. Aguarde até que a distribuição alcance o estado “Implantado” antes de testar.

Problemas de reprodução e inserção de anúncios

O conteúdo é reproduzido, mas nenhum anúncio é inserido

Causas possíveis:

- O servidor de decisão de anúncios não está respondendo ou retorna um VAST vazio
- O conteúdo não contém marcadores de anúncios

Solução: verifique se seu servidor de anúncios está acessível e está retornando respostas VAST válidas. Verifique se seu conteúdo tem marcadores de anúncio adequados (marcadores SCTE-35 para conteúdo ao vivo ou tags de quebra de anúncio para VOD).

403 Erros proibidos ao acessar o conteúdo

Causas possíveis:

- O controle de acesso do Origin não está configurado corretamente
- Problemas de permissões de bucket ou endpoint do Origin

Solução: Para origens do Amazon S3, verifique se a política de bucket permite acesso a partir da CloudFront distribuição. Para MediaPackage origens, verifique se o controle de acesso de origem está configurado corretamente e se o endpoint está acessível.

Erros de reprodução ou armazenamento em buffer

Causas possíveis:

- Os padrões de caminho de comportamento do cache não correspondem aos caminhos de conteúdo
- Configuração incorreta do domínio de origem

Solução: verifique se seus comportamentos de cache têm os padrões de caminho corretos para encaminhar as solicitações para as origens apropriadas. Verifique CloudFront os registros para ver qual origem está lidando com as solicitações e confirme se é a esperada.

Para profissionais de radiodifusão, essas dicas adicionais de solução de problemas podem ajudar:

- Use o Amazon CloudWatch Logs Insights para pesquisar padrões de erro específicos nos MediaTailor registros
- Teste primeiro com um servidor de anúncios VAST simples (como o padrão fornecido no modelo) antes de usar seu servidor de anúncios de produção

- Verifique os marcadores de anúncios do seu conteúdo usando a ferramenta de inspeção de MediaTailor manifestos no console
- Verifique o tráfego de rede nas ferramentas de desenvolvedor do seu navegador para ver se as solicitações ao servidor de anúncios estão sendo feitas corretamente

Para solucionar problemas adicionais, verifique os CloudWatch registros MediaTailor e CloudFront identifique erros específicos.

Personalize o CloudFormation modelo para CDN e integrações MediaTailor

AWS Elemental MediaTailor a personalização do modelo permite que os profissionais de radiodifusão adaptem o AWS CloudFormation modelo para atender aos requisitos específicos do fluxo de trabalho. Embora o modelo básico funcione para muitos cenários, essas personalizações podem ajudá-lo a atender a necessidades mais complexas.

Os exemplos abaixo mostram trechos de código YAML que você pode adicionar ao modelo. Se você não estiver familiarizado com o YAML ou a CloudFormation sintaxe, considere trabalhar com um desenvolvedor ou arquiteto de AWS soluções para fazer essas alterações.

Você pode personalizar o CloudFormation modelo para atender aos requisitos específicos do fluxo de trabalho.

Adicionar ou modificar origens

Para fluxos de trabalho de transmissão que usam várias fontes de conteúdo (como fontes primárias e de backup, ou bibliotecas de conteúdo diferentes), você pode adicionar outras origens à sua CloudFront distribuição:

```
Origins:
  # Add a new origin for additional content
  - Id: SecondaryContentOrigin
    DomainName: secondary-content.example.com
    CustomOriginConfig:
      OriginProtocolPolicy: 'https-only'
      OriginSSLProtocols:
        - TLSv1.2
```

Em seguida, adicione um comportamento de cache correspondente para rotear padrões específicos para essa origem:

CacheBehaviors:

```
- PathPattern: '/secondary-content/*'
  TargetOriginId: SecondaryContentOrigin
  ViewerProtocolPolicy: 'https-only'
  CachePolicyId: 658327ea-f89d-4fab-a63d-7e88639e58f6 # Managed-CachingOptimized
```

Crie políticas de cache personalizadas

Para fluxos de trabalho de transmissão com requisitos específicos de armazenamento em cache (como parâmetros de seleção de qualidade ou autenticação do visualizador), você pode criar políticas de cache personalizadas em vez de usar as gerenciadas. Para obter orientação detalhada sobre valores de TTL e estratégias de armazenamento em cache, consulte [Otimização de cache para CDN e integrações MediaTailor](#)

```
# Define a custom cache policy
CustomCachePolicy:
  Type: AWS::CloudFront::CachePolicy
  Properties:
    CachePolicyConfig:
      Name: !Sub '${AWS::StackName}-CustomCachePolicy'
      DefaultTTL: 86400 # 24 hours
      MaxTTL: 31536000 # 1 year
      MinTTL: 1 # 1 second
      ParametersInCacheKeyAndForwardedToOrigin:
        CookiesConfig:
          CookieBehavior: none
        HeadersConfig:
          HeaderBehavior: none
        QueryStringsConfig:
          QueryStringBehavior: whitelist
          QueryStrings:
            - quality
            - format

# Reference the custom policy in a cache behavior
CacheBehaviors:
  - PathPattern: '/custom-path/*'
    TargetOriginId: ContentOrigin
    ViewerProtocolPolicy: 'https-only'
    CachePolicyId: !Ref CustomCachePolicy
```


Aprimorar a MediaTailor configuração

Para fluxos de trabalho de transmissão que precisam de recursos avançados de inserção de anúncios, você pode aprimorar a MediaTailor configuração com opções como busca prévia de anúncios (para reduzir a latência), limites de personalização e anúncios bumper.

```
MediaTailorPlaybackConfig:
  Type: AWS::MediaTailor::PlaybackConfiguration
  Properties:
    # Add ad prefetching for improved performance
    AvailSuppression:
      Mode: BEHIND_LIVE_EDGE
      Value: 00:00:00
    # Add personalization parameters
    PersonalizationThresholdSeconds: 2
    # Add bumper ads
    Bumper:
      StartUrl: https://example.com/bumper-start.mp4
      EndUrl: https://example.com/bumper-end.mp4
    # Other existing properties...
```

Para obter mais informações sobre MediaTailor as opções de configuração, consulte [Utilizar AWS Elemental MediaTailor para inserir anúncios](#).

Adicione recursos de segurança

Para fluxos de trabalho de transmissão com requisitos de segurança específicos (como restrições geográficas ou proteção contra ataques DDoS), você pode adicionar AWS WAF integração e restrições geográficas:

```
# Create a AWS WAF Web ACL
WebACL:
  Type: AWS::WAFv2::WebACL
  Properties:
    Name: !Sub '${AWS::StackName}-WebACL'
    Scope: CloudFront
    DefaultAction:
      Allow: {}
    VisibilityConfig:
      SampledRequestsEnabled: true
      CloudWatchMetricsEnabled: true
```

```
    MetricName: !Sub '${AWS::StackName}-WebACL '
Rules:
  - Name: RateLimitRule
    Priority: 0
    Action:
      Block: {}
    VisibilityConfig:
      SampledRequestsEnabled: true
      CloudWatchMetricsEnabled: true
      MetricName: RateLimitRule
    Statement:
      RateBasedStatement:
        Limit: 1000
        AggregateKeyType: IP

# Reference the AWS WAF Web ACL in the CloudFront distribution
CloudFrontDistribution:
  Type: AWS::CloudFront::Distribution
  Properties:
    DistributionConfig:
      WebACLId: !GetAtt WebACL.Arn
      # Add geo-restriction
      Restrictions:
        GeoRestriction:
          RestrictionType: whitelist
          Locations:
            - US
            - CA
            - GB
      # Other existing properties...
```

Para obter mais informações sobre CloudFormation modelos, consulte o [Guia AWS CloudFormation do usuário](#).

Para CloudFormation modelos e exemplos específicos de transmissão, consulte o repositório de ferramentas de [serviços de mídia da AWS](#). GitHub

AWS CloudFormation referência de modelo AWS Elemental MediaTailor e CloudFront integração com a Amazon

AWS Elemental MediaTailor a integração com a Amazon CloudFront pode ser automatizada usando o seguinte AWS CloudFormation modelo completo:

AWSTemplateFormatVersion: '2010-09-09'

Description: | CloudFormation template that sets up AWS Elemental MediaTailor integration with CloudFront Distribution for server-side ad insertion. This template supports various content origins including MediaPackage, Amazon S3, and custom origins, making it versatile for different streaming architectures.

Parameters:

AdServerUrl:

Type: String

Default: 'https://d1kbnmkziz9rksx.CloudFront.net/VASTEndpoint.xml'

Description: URL of the VAST ad server for dynamic ad insertion. Static VAST endpoint provided for testing.

ContentOriginDomainName:

Type: String

Description: |

Domain name of your content origin without protocol (e.g., *mediapackage-domain.mediapackagev2.us-west-2.amazonaws.com*, *mybucket.s3.amazonaws.com*, or *custom-origin.example.com*). Do not include http:// or https:// prefixes or any paths.

AllowedPattern: "^(([a-zA-Z0-9]|[a-zA-Z0-9][a-zA-Z0-9\\-]*[a-zA-Z0-9])\\.)*([A-Za-z0-9]|[A-Za-z0-9][A-Za-z0-9\\-]*[A-Za-z0-9])\$"

ConstraintDescription: Must be a valid domain name (e.g., *example.com*) without protocol or path components. IP addresses are not allowed.

ContentOriginType:

Type: String

AllowedValues:

- mediapackagev2
- s3
- custom

Default: mediapackagev2

Description: |

The type of content origin:

- mediapackagev2: AWS Elemental MediaPackage V2
- s3: Amazon S3 bucket
- custom: Any other custom origin

Resources:

```
#-----
# Origin Access Control (for securing MediaPackage V2 and Amazon S3 origins)
#-----
```

```

CloudFrontOriginAccessControl:
  Type: AWS::CloudFront::OriginAccessControl
  Condition: IsNotCustomOrigin
  Properties:
    OriginAccessControlConfig:
      Name: !Sub '${AWS::StackName}-OAC'
      OriginAccessControlOriginType: !Ref ContentOriginType
      SigningBehavior: always
      SigningProtocol: sigv4
      Description: Origin Access Control for content origin

#-----
# MediaTailor Playback Configuration
#-----
MediaTailorPlaybackConfig:
  Type: AWS::MediaTailor::PlaybackConfiguration
  Properties:
    Name: !Sub '${AWS::StackName}-PlaybackConfig'
    # The video content source should point to your CloudFront distribution
    VideoContentSourceUrl: !Sub 'https://${CloudFrontDistribution.DomainName}/'
    # The Ad Decision Server URL is where MediaTailor will request ads
    AdDecisionServerUrl: !Ref AdServerUrl
    # Configuration for pre-roll ads during live streams
    LivePreRollConfiguration:
      AdDecisionServerUrl: !Ref AdServerUrl
      MaxDurationSeconds: 30
    # CDN configuration for integrating with CloudFront
    CdnConfiguration:
      AdSegmentUrlPrefix: '/'
      ContentSegmentUrlPrefix: '/'
    # Set a reasonable manifest segment timeout
    ManifestProcessingRules:
      AdMarkerPassthrough:
        Enabled: false

#-----
# CloudFront Distribution
#-----
CloudFrontDistribution:
  Type: AWS::CloudFront::Distribution
  Properties:
    DistributionConfig:
      Enabled: true
      HttpVersion: http2and3

```

```

    IPV6Enabled: true
    Comment: !Sub 'Distribution for MediaTailor ad insertion with
    ${ContentOriginType} origin'

    # Default cache behavior points to the content origin
    DefaultCacheBehavior:
      TargetOriginId: ContentOrigin
      ViewerProtocolPolicy: 'https-only'
      # Using managed policies for optimal performance and simplicity
      CachePolicyId: 658327ea-f89d-4fab-a63d-7e88639e58f6 # Managed-
CachingOptimized
      OriginRequestPolicyId: 88a5eaf4-2fd4-4709-b370-b4c650ea3fcf # Managed-
HostHeaderOnly
      ResponseHeadersPolicyId: eaab4381-ed33-4a86-88ca-d9558dc6cd63 # Managed-
CORS-with-preflight-and-SecurityHeadersPolicy
      Compress: true

    # Define all the origins needed for the workflow
    Origins:
      # Main content origin (MediaPackage, Amazon S3, or Custom)
      - Id: ContentOrigin
        DomainName: !Ref ContentOriginDomainName
        # Apply Origin Access Control for secure origins
        OriginAccessControlId: !If [IsNotCustomOrigin, !GetAtt
CloudFrontOriginAccessControl.Id, !Ref "AWS::NoValue"]
        # For custom origins, we need a CustomOriginConfig
        CustomOriginConfig:
          OriginProtocolPolicy: 'https-only'
          OriginSSLProtocols:
            - TLSv1.2
          OriginKeepaliveTimeout: 5
          OriginReadTimeout: 30
          HTTPPort: 80
          HTTPSPort: 443

      # MediaTailor Manifests Origin - handles manifest manipulation for ad
insertion
      - Id: MediaTailorManifests
        DomainName: !Sub 'manifests.mediataylor.${AWS::Region}.amazonaws.com'
        CustomOriginConfig:
          OriginProtocolPolicy: 'https-only'
          OriginSSLProtocols:
            - TLSv1.2
          OriginKeepaliveTimeout: 5

```

```

        OriginReadTimeout: 30
    # Origin Shield improves caching efficiency
    OriginShield:
        Enabled: true
        OriginShieldRegion: !Ref AWS::Region

# MediaTailor Segments Origin - handles personalized ads
- Id: MediaTailorSegments
  DomainName: !Sub 'segments.mediatailor.${AWS::Region}.amazonaws.com'
  CustomOriginConfig:
    OriginProtocolPolicy: 'https-only'
    OriginSSLProtocols:
      - TLSv1.2
    OriginKeepaliveTimeout: 5
    OriginReadTimeout: 30

# Cache behaviors to route specific request patterns to the right origin
CacheBehaviors:
  # Handle MediaTailor segment requests for ad content which are cache-able
  - PathPattern: '/tm/*'
    TargetOriginId: MediaTailorSegments
    ViewerProtocolPolicy: 'https-only'
    CachePolicyId: 658327ea-f89d-4fab-a63d-7e88639e58f6 # Managed-
CachingOptimized
    OriginRequestPolicyId: 88a5eaf4-2fd4-4709-b370-b4c650ea3fcf # Managed-
HostHeaderOnly
    ResponseHeadersPolicyId: eaab4381-ed33-4a86-88ca-d9558dc6cd63 # Managed-
CORS-with-preflight-and-SecurityHeadersPolicy
    Compress: true

  # Handle MediaTailor interstitial (SGAI) media requests which are cache-able
  - PathPattern: '/v1/i-media/*'
    TargetOriginId: MediaTailorManifests
    ViewerProtocolPolicy: 'https-only'
    CachePolicyId: 658327ea-f89d-4fab-a63d-7e88639e58f6 # Managed-
CachingOptimized
    OriginRequestPolicyId: 88a5eaf4-2fd4-4709-b370-b4c650ea3fcf # Managed-
HostHeaderOnly
    ResponseHeadersPolicyId: eaab4381-ed33-4a86-88ca-d9558dc6cd63 # Managed-
CORS-with-preflight-and-SecurityHeadersPolicy
    Compress: true

  # Handle MediaTailor Personalized manifests which are not cache-able
  - PathPattern: '/v1/*'

```

```

    TargetOriginId: MediaTailorManifests
    ViewerProtocolPolicy: 'https-only'
    CachePolicyId: 4135ea2d-6df8-44a3-9df3-4b5a84be39ad # Managed-
CachingDisabled
    OriginRequestPolicyId: 59781a5b-3903-41f3-afcb-af62929ccde1 # Managed-
AllViewer
    ResponseHeadersPolicyId: eaab4381-ed33-4a86-88ca-d9558dc6cd63 # Managed-
CORS-with-preflight-and-SecurityHeadersPolicy
    Compress: true

    # Handle MediaTailor segment *redirect* requests which are not cache-able
    (used for server side reporting)
    - PathPattern: '/segment/*'
      TargetOriginId: MediaTailorManifests
      ViewerProtocolPolicy: 'https-only'
      CachePolicyId: 4135ea2d-6df8-44a3-9df3-4b5a84be39ad # Managed-
CachingDisabled
      OriginRequestPolicyId: 59781a5b-3903-41f3-afcb-af62929ccde1 # Managed-
AllViewer
      ResponseHeadersPolicyId: eaab4381-ed33-4a86-88ca-d9558dc6cd63 # Managed-
CORS-with-preflight-and-SecurityHeadersPolicy
      Compress: true

Conditions:
  IsNotCustomOrigin: !Not [!Equals [!Ref ContentOriginType, 'custom']]

Outputs:
  CloudFrontDomainName:
    Description: Domain name of the CloudFront distribution
    Value: !GetAtt CloudFrontDistribution.DomainName

  HlsManifestUrl:
    Description: URL for HLS manifest with ads inserted (append your manifest path)
    Value: !Sub 'https://
${CloudFrontDistribution.DomainName}${MediaTailorPlaybackConfig.HlsConfiguration.ManifestEndpoint

  DashManifestUrl:
    Description: URL for DASH manifest with ads inserted (append your manifest path)
    Value: !Sub 'https://
${CloudFrontDistribution.DomainName}${MediaTailorPlaybackConfig.DashConfiguration.ManifestEndpoint

  MediaTailorPlaybackConfigName:
    Description: Name of the MediaTailor playback configuration

```

```
Value: !Ref MediaTailorPlaybackConfig
```

Production-ready CloudFront configuração para MediaTailor

Essa configuração CloudFront de distribuição fornece tudo o que você precisa para fornecer MediaTailor conteúdo com inserção de anúncios no lado do servidor em grande escala. Copie essa configuração e personalize-a de acordo com suas origens e requisitos específicos.

O que essa configuração realiza

Essa configuração cria uma CloudFront distribuição pronta para produção que lida com todos os tipos de MediaTailor solicitações com armazenamento em cache e desempenho ideais. Ele inclui três origens (seu conteúdo, MediaTailor segmentos e MediaTailor manifestos) com quatro comportamentos de cache que encaminham as solicitações corretamente e armazenam o conteúdo em cache de forma adequada.

Quando usar essa configuração

Essa configuração é ideal para streaming ao vivo, vídeo sob demanda e fluxos de trabalho híbridos que exigem inserção de anúncios no servidor.

Three-origin arquitetura

MediaTailor usa um padrão de arquitetura de três origens para otimizar a entrega de conteúdo e o desempenho de inserção de anúncios. Cada origem tem uma finalidade específica no fluxo de trabalho de inserção de anúncios:

A origem do seu conteúdo

Essa é a verdadeira origem do seu conteúdo que alimenta MediaTailor. Por exemplo, isso pode ser AWS Elemental MediaPackage V2 ou outro serviço de entrega de conteúdo. Essa origem veicula seu conteúdo original antes da inserção do anúncio. Os exemplos comuns incluem:

- MediaPackage Configurações de empacotamento V2
- Third-party redes de entrega de conteúdo
- On-premises servidores de streaming
- Buckets Amazon S3 com conteúdo estático

MediaTailor origem dos segmentos

Essa origem usa o nome do host `segments.mediatailor.region.amazonaws.com` e veicula os segmentos de anúncios reais depois MediaTailor de codificá-los. Esses são os segmentos de vídeo que contêm os anúncios. Essa origem trata de:

- Segmentos de anúncios transcodificados no mesmo formato do seu conteúdo
- Solicitações redirecionadas do padrão de `/segment/*` caminho
- Segmentos de anúncios que foram processados para inserção de anúncios no servidor

MediaTailor manifesta origem

Essa origem usa o nome do host `manifests.mediatailor.region.amazonaws.com` e pode ser usada como um nome de host regional para configurações de reprodução na região especificada. AWS MediaTailor seleciona a configuração de reprodução correta com base no caminho da solicitação. Essa origem fornece:

- Manifestos HLS e DASH personalizados com inserção de anúncios específicos para o espectador
- Server-guided manifestos de inserção de anúncios (SGAI) para conteúdo armazenável em cache
- Rastreamento de anúncios e tratamento de beacons para relatórios do lado do servidor

Com o nome do host de origem `manifests.mediatailor.region.amazonaws.com`, você pode ter várias configurações de reprodução que funcionam com a mesma CloudFront distribuição se estiverem na região especificada e incluir o nome da configuração de reprodução no caminho da solicitação. Por exemplo:

- `https://your-distribution.cloudfront.net/v1/master/playback-config-1/manifest.m3u8`
- `https://your-distribution.cloudfront.net/v1/master/playback-config-2/manifest.m3u8`

Selecione as políticas de solicitação de origem com base no tipo de conteúdo para evitar o envenenamento do cache e, ao mesmo tempo, garantir a funcionalidade adequada. A principal distinção é entre conteúdo que pode ser armazenado em cache e não pode ser armazenado em cache:

- Manifestos (não armazenáveis em cache): use AllViewer para encaminhar todos os cabeçalhos necessários para conteúdo dinâmico. Como os manifestos não são armazenados em cache, não há risco de envenenamento do cache.
- Segmentos (armazenáveis em cache): use None para obter um desempenho ideal.
- Origens do S3: use CORS-S3Origin para buckets do Amazon S3
- MediaPackage origens: use CORS-S3Origin para endpoints MediaPackage V2

Origins						
Filter origins by property or value						
	Origin name	Origin domain	Origin path	Origin type	Origin Shield region	Origin access
☐	ContentOrigin	zrljic.egress.kv9ydh.mediapackagev2.us-west-2.amazonaws.com		MediaPackage V2	us-west-2	E2SRJY3NEMC428
☐	MediaTailorSegments	segments.mediatailor.us-west-2.amazonaws.com		MediaTailor	-	-
☐	MediaTailorManifests	manifests.mediatailor.us-west-2.amazonaws.com		MediaTailor	us-west-2	-

Precedência e configuração do comportamento do cache

MediaTailor requer configurações específicas de comportamento de cache para lidar adequadamente com diferentes tipos de solicitações. A precedência dos comportamentos de cache é fundamental porque as CDNs os processam em ordem (do menor para o maior) e usam o comportamento para o primeiro padrão de caminho correspondente. Compreender essa precedência é essencial para solucionar problemas:

- Precedência 0: os padrões mais específicos (como /tm/*) são avaliados primeiro
- Números de precedência mais altos: padrões menos específicos são avaliados em ordem
- Comportamento padrão: captura todas as solicitações que não correspondem a outros padrões

Se as solicitações não estiverem se comportando conforme o esperado, verifique se seus padrões de caminho não se sobrepõem de forma não intencional.

General

Security

Origins

Behaviors

Error pages

Invalidations

Tags

Logging

Behaviors

Save

Move up

Move down

Edit

Delete

Create behavior

Filter behaviors by property or value

	Preced...	Path patt...	Origin or origin group	Viewer pr...	Cache policy name	Origin request policy name	Response headers policy name
☐	0	/tm/*	MediaTailorSegments	HTTPS only	Managed-CachingOptimized	Managed-CORS-S3Origin	Managed-CORS-with-preflight-and-SecurityHeadersPolicy
☐	1	/i-media/*	MediaTailorManifests	HTTPS only	Managed-CachingOptimized	Managed-CORS-S3Origin	Managed-CORS-with-preflight-and-SecurityHeadersPolicy
☐	2	/v1/*	MediaTailorManifests	HTTPS only	Managed-CachingDisabled	Managed-AllViewerAndCloudFrontHeaders-2022-06	Managed-CORS-with-preflight-and-SecurityHeadersPolicy
☐	3	/segment/*	MediaTailorManifests	HTTPS only	Managed-CachingDisabled	Managed-AllViewerAndCloudFrontHeaders-2022-06	Managed-CORS-with-preflight-and-SecurityHeadersPolicy
☐	4	Default (*)	ContentOrigin	HTTPS only	Managed-CachingOptimized	CustomOriginRequestPolicyWithWithLLHLSDDirectives	Managed-CORS-with-preflight-and-SecurityHeadersPolicy

Precedência 0: comportamento do caminho dos segmentos de anúncios

Esse comportamento lida com as solicitações redirecionadas do comportamento do caminho do segmento, veiculando os segmentos de anúncios reais. CloudFront aplica os seguintes comportamentos a todas as solicitações com um padrão de `/tm/*` caminho. Esse é o comportamento de maior prioridade porque a entrega do segmento de anúncios é fundamental para a reprodução ininterrupta.

- Padrão de caminho: `/tm/*`

Exemplos de URLs que correspondem a esse padrão:

- `https://your-distribution.cloudfront.net/tm/ad-segment-001.ts`
- `https://your-distribution.cloudfront.net/tm/transcoded-ad.m4s`
- Origem: a origem que você criou com o `segments.mediatailor.region.amazonaws.com` domínio.

Isso está MediaTailorSegments no exemplo da seção anterior sobre origens.

- Política de cache: `Managed-CachingOptimized`

Para obter detalhes sobre o que está incluído na política de cache CloudFront gerenciado, consulte [CachingOptimized](#) no guia CloudFront do usuário. Você também pode usar essas mesmas configurações da política gerenciada em sua CDN de terceiros.

- Política de solicitação de origem: `None`
- Política de cabeçalhos de resposta: `Managed-CORS-with-preflight-and-SecurityHeadersPolicy`

Para obter detalhes sobre o que está incluído na política de cabeçalhos de resposta, consulte [CORS-with-preflight-and-SecurityHeadersPolicy](#) no guia CloudFront do usuário.

Adaptando essas configurações a outras CDNs

Se você estiver usando uma CDN diferente de CloudFront, procure configurações equivalentes que realizem o seguinte.

Combinação de padrões de caminho

Configure um comportamento específico para o padrão de `/tm/*` caminho para lidar com segmentos de MediaTailor anúncios

Configuração da chave de cache

Inclua o `Origin` cabeçalho em sua chave de cache para garantir que as respostas sejam armazenadas em cache separadamente para origens diferentes

Encaminhamento de cabeçalho

Encaminhar o `Origin` cabeçalho e outros CORS-related cabeçalhos para a origem

Gerenciamento do cabeçalho de resposta

Configure sua CDN para garantir que o `Access-Control-Allow-Origin` cabeçalho esteja presente nas respostas

A terminologia específica e as opções de configuração variam de acordo com o provedor de CDN, mas os princípios subjacentes permanecem os mesmos.

Precedência 1: comportamento de inserção de Server-guided anúncios

Esse comportamento processa solicitações [MediaTailor visão geral e implementação da inserção de anúncios guiada pelo servidor](#) (SGAI) quando os clientes configuram o modo guiado, que fornece manifestos que podem ser armazenados em cache. CloudFront aplica os seguintes comportamentos a todas as solicitações com um padrão de `/v1/i-media/*` caminho. O SGAi permite um melhor desempenho de cache porque os manifestos não são específicos para o visualizador.

- Padrão de caminho: `/v1/i-media/*` (caminho iMedia para SGAi)

Exemplos de URLs que correspondem a esse padrão:

- `https://your-distribution.cloudfront.net/v1/i-media/your-config/manifest.m3u8`
- `https://your-distribution.cloudfront.net/v1/i-media/your-config/playlist.mpd`
- Origem: a origem que você criou com o `manifests.mediatailor.region.amazonaws.com` domínio.

Isso está MediaTailorManifests no exemplo da seção anterior sobre origens.

- Política de cache: `Managed-CachingOptimized`

Para obter detalhes sobre o que está incluído na política de cache CloudFront gerenciado, consulte [CachingOptimized](#)o guia CloudFront do usuário. Você também pode usar essas mesmas configurações da política gerenciada em sua CDN de terceiros.

- Política de solicitação de origem: None
- Política de cabeçalhos de resposta: Managed-CORS-with-preflight-and-SecurityHeadersPolicy

Para obter detalhes sobre o que está incluído na política de cabeçalhos de resposta, consulte [CORS-with-preflight-and-SecurityHeadersPolicy](#)o guia CloudFront do usuário.

Precedência 2: comportamento manifesto personalizado

Esse comportamento lida com solicitações de manifesto personalizadas. CloudFront aplica os seguintes comportamentos a todas as solicitações com um padrão de `/v1/*` caminho. CloudFront aplica os seguintes comportamentos e não armazena em cache manifestos personalizados porque eles contêm URLs de conteúdo de anúncio específicos do espectador. Esses comportamentos se aplicam a todas as solicitações que têm um padrão de `/v1/*` caminho. Essa é a principal MediaTailor funcionalidade em que cada espectador recebe um manifesto exclusivo com inserção de anúncios personalizados.

- Padrão de caminho: `/v1/*` (MediaTailor solicitações V1 padrão)

Exemplos de URLs que correspondem a esse padrão:

- `https://your-distribution.cloudfront.net/v1/master/your-config/manifest.m3u8`
- `https://your-distribution.cloudfront.net/v1/dash/your-config/manifest.mpd`
- Origem: a origem que você criou com o `manifests.mediatailor.region.amazonaws.com` domínio.

Isso está MediaTailorManifestsno exemplo da seção anterior sobre origens.

- Política de cache: Managed-CachingDisabled

Para obter detalhes sobre o que está incluído na política de cache, consulte [CachingDisabled](#)o guia CloudFront do usuário.

- Política de solicitação de origem: AllViewer

Para manifestos personalizados, use a AllViewer política para encaminhar todos os cabeçalhos necessários para conteúdo dinâmico.

- Política de cabeçalhos de resposta: Managed-CORS-with-preflight-and-SecurityHeadersPolicy

Para obter detalhes sobre o que está incluído na política de cabeçalhos de resposta, consulte [CORS-with-preflight-and-SecurityHeadersPolicy](#)o guia CloudFront do usuário.

Accept-Encoding cabeçalho

Recomendamos que sua CDN preserve o Accept-Encoding cabeçalho dos visualizadores. Esse cabeçalho fornece MediaTailor instruções sobre como compactar manifestos personalizados.

Em CloudFront, a política de solicitação de AllViewerAndCloudFrontHeaders origem inclui a passagem do Accept-Encoding cabeçalho do visualizador. Se você usar uma CDN diferente, certifique-se de que ela preserve esse cabeçalho.

Veja a seguir como MediaTailor manipula o Accept-Encoding cabeçalho.

- Dispositivos antigos: TVs inteligentes mais antigas que não suportam gzip não enviam o Accept-Encoding cabeçalho, então MediaTailor retornam manifestos não compactados
- Dispositivos modernos: iPhones, navegadores Chrome e outros clientes modernos enviam o Accept-Encoding cabeçalho, permitindo MediaTailor compactar os manifestos antes da entrega

Precedência 3: Server-side comportamento do caminho do farol

Esse comportamento manipula solicitações MediaTailor que resultam em redirecionamentos para. [Server-side rastreamento](#) Essas solicitações são essenciais para rastrear beacons, portanto, cada solicitação deve ser processada por. MediaTailor CloudFront aplica os seguintes comportamentos a todas as solicitações com um padrão de /segment/* caminho.

- Padrão de caminho: /segment/*

Exemplos de URLs que correspondem a esse padrão:

- <https://your-distribution.cloudfront.net/segment/tracking-beacon-123>
- <https://your-distribution.cloudfront.net/segment/ad-request-456.ts>

- Origem: a origem que você criou com o `manifests.mediatailor.region.amazonaws.com` domínio.

Isso está `MediaTailorManifests` no exemplo da seção anterior sobre origens.

- Política de cache: `Managed-CachingDisabled`

Para obter detalhes sobre o que está incluído na política de cache, consulte [CachingDisabled](#) o guia CloudFront do usuário.

- Política de solicitação de origem: `AllViewer`

Para solicitações de beacon do lado do servidor, use a `AllViewer` política para encaminhar todos os cabeçalhos necessários para o rastreamento. Como essas solicitações não são armazenadas em cache, não há risco de envenenamento do cache.

- Política de cabeçalhos de resposta: `Managed-CORS-with-preflight-and-SecurityHeadersPolicy`

Para obter detalhes sobre o que está incluído na política de cabeçalhos de resposta, consulte [CORS-with-preflight-and-SecurityHeadersPolicy](#) o guia CloudFront do usuário.

Quando MediaTailor processa essas solicitações, ele retorna uma resposta de redirecionamento 302 com um caminho que aponta para a localização real do segmento. Por exemplo, uma solicitação para `/segment/ad123.ts` pode redirecionar para a `/tm/encoded-ad-segment.ts` origem do segmento.

Precedência 4: comportamento do caminho de origem do conteúdo

Se o caminho da solicitação não corresponder a nenhum dos outros padrões, CloudFront aplica o comportamento padrão. Esse comportamento envia solicitações diretamente para a origem do conteúdo, sem processamento de MediaTailor. Isso permite acesso direto à origem do seu conteúdo (como `MediaPackage V2`) quando necessário. CloudFront aplica os seguintes comportamentos a todas as solicitações que não incluem nenhum dos padrões de caminho anteriores.

- Padrão de caminho: `(*)`
- Origem: a origem que você criou com o domínio da origem do seu conteúdo.

Isso está `ContentOrigin` no exemplo da seção anterior sobre origens.

- Política de cache: `Managed-CachingOptimized`

Para obter detalhes sobre o que está incluído na política de cache CloudFront gerenciado, consulte [CachingOptimized](#) no guia CloudFront do usuário. Você também pode usar essas mesmas configurações da política gerenciada em sua CDN de terceiros.

 Note

Para implementações de HLS de baixa latência, considere usar uma política de cache personalizada com diretivas HLS de baixa latência (LLH) em vez da política padrão. `CachingOptimized`

- Política de solicitação de origem: None
- Política de cabeçalhos de resposta: Managed-CORS-with-preflight-and-SecurityHeadersPolicy

Embora o comportamento padrão da origem do conteúdo geralmente não enfrente os mesmos riscos de envenenamento do cache do CORS que o comportamento do segmento de anúncios, ainda é recomendável usar a política de cabeçalhos de Managed-CORS-with-preflight-and-SecurityHeadersPolicy resposta e incluir o `Origin` cabeçalho na sua chave de cache. Isso garante o tratamento consistente do CORS em todos os tipos de conteúdo e evita possíveis problemas de reprodução em reprodutores baseados na web.

Para segmentos de conteúdo, a política de Managed-CachingOptimized cache fornece bom desempenho, enquanto a política de cabeçalhos de Managed-CORS-with-preflight-and-SecurityHeadersPolicy resposta garante o tratamento adequado do CORS. Essa combinação permite um armazenamento em cache eficiente, mantendo a compatibilidade com players baseados na web que exigem cabeçalhos CORS.

A aplicação consistente do CORS em segmentos de anúncios e segmentos de conteúdo cria uma experiência de reprodução mais confiável e simplifica a solução de problemas. Sem a configuração adequada do CORS, os jogadores podem ter um comportamento inconsistente ao fazer a transição entre conteúdo e anúncios.

Para obter detalhes sobre o que está incluído na política de cabeçalhos de resposta, consulte [CORS-with-preflight-and-SecurityHeadersPolicy](#) no guia CloudFront do usuário.

Obtenha suporte e ajuda para solução de problemas para CDN e integrações MediaTailor

AWS Elemental MediaTailor Os problemas de integração da CDN podem ser complexos de diagnosticar e resolver. Use este guia quando precisar de ajuda adicional com problemas de CDN e MediaTailor integração que não podem ser resolvidos por meio da solução de problemas de autoatendimento.

Antes de escalar, experimente estas opções de autoatendimento:

- Siga as etapas de solução de problemas em [Solucionar problemas de integração com a CDN](#)
- Analise seus registros e códigos de erro usando [Referência de análise de log de integração CDN](#)
- Revise sua configuração de monitoramento com [Configurar ferramentas de monitoramento](#)
- Consulte o [guia MediaTailor de solução de problemas](#) para problemas específicos do serviço
- Pesquise no [AWS re:Post](#) problemas semelhantes e soluções comunitárias
- Consulte o [AWS Knowledge Center](#) para conhecer padrões de integração comuns

Tópicos

- [Quando escalar problemas de MediaTailor CDN para o AWS Support](#)
- [Reúna informações MediaTailor da CDN antes de entrar em contato com o suporte](#)
- [MediaTailor Recursos e canais de suporte à integração CDN](#)
- [MediaTailor Melhores práticas de casos de suporte de integração de CDN](#)

Quando escalar problemas de MediaTailor CDN para o AWS Support

AWS Elemental MediaTailor Os problemas de integração da CDN devem ser encaminhados para o AWS Support quando a solução de problemas de autoatendimento não resolver o problema.

Considere recorrer ao AWS Support quando:

Observação: o suporte técnico exige um plano de AWS suporte pago. Para obter detalhes do plano de suporte e tempos de resposta, consulte os [planos do AWS Support](#).

- Os problemas afetam o tráfego de produção ou a receita
- Você seguiu todas as etapas de solução de problemas sem resolução

- Os padrões de erro sugerem problemas de nível de serviço
- Você precisa de ajuda com cenários complexos de configuração
- Os problemas de desempenho persistem apesar dos esforços de otimização

Reúna informações MediaTailor da CDN antes de entrar em contato com o suporte

AWS Elemental MediaTailor Os casos de suporte de integração de CDN exigem informações específicas de configuração e diagnóstico para uma solução eficaz de problemas. Antes de entrar em contato com o AWS Support, reúna essas informações para ajudar a agilizar a resolução:

Dica: ter essas informações prontas antes de criar seu caso de suporte reduzirá significativamente o tempo de resolução e ajudará os engenheiros de suporte a entender sua configuração específica.

- MediaTailor detalhes da configuração:
 - Nome da configuração e AWS região
 - ARN de configuração de reprodução
 - URL do ADS e tipo de integração
- Detalhes da configuração do CDN:
 - ID de distribuição CDN ou nome da configuração
 - Configurações de comportamento de cache para manifestos e segmentos
 - Configuração de origem e regras de roteamento
- Informações de erro:
 - Mensagens de erro específicas e códigos de status HTTP
 - Registros de data e hora quando ocorrem problemas
 - Amostra URLs que demonstra o problema
 - CDN e entradas de MediaTailor registro relacionadas ao problema
- Informações de teste:
 - Etapas que você já tomou para solucionar o problema
 - Dispositivos e players em que o problema ocorre
 - Se o problema afeta todo o conteúdo ou transmissões específicas
 - Frequência e padrão do problema (intermitente, consistente, baseado no tempo)

MediaTailor Recursos e canais de suporte à integração CDN

AWS Elemental MediaTailor O suporte à integração de CDN está disponível por meio de vários canais para ajudá-lo a resolver problemas e otimizar sua implementação:

- AWS Support: crie um caso de suporte por meio do AWS Management Console para obter assistência técnica

Acesso: [AWS Support Center](#)

- AWS re:POST: plataforma de perguntas e respostas orientada pela comunidade para perguntas relacionadas e suporte comunitário AWS

Acesso: [AWS Re:post](#)

- AWS Documentação: guias abrangentes MediaTailor e serviços de CDN

Acesso: [MediaTailorDocumentação](#) e [CloudFrontdocumentação](#)

- AWS Treinamento: cursos e certificações para serviços de mídia e otimização de CDN

Acesso: [AWS Treinamento e certificação](#)

- AWS Centro de conhecimento: artigos selecionados para AWS problemas comuns e melhores práticas

Acesso: [Centro de AWS conhecimento](#)

- AWS Trusted Advisor: recomendações automatizadas para otimização e melhores práticas

Acesso: [AWS Trusted Advisor](#)

- AWS Personal Health Dashboard: notificações de integridade e manutenção do serviço

Acesso: [AWS Personal Health Dashboard](#)

MediaTailor Melhores práticas de casos de suporte de integração de CDN

AWS Elemental MediaTailor Os casos de suporte de integração de CDN são resolvidos com mais eficiência quando você segue essas melhores práticas. Para obter a resolução mais rápida:

- Escolha o nível de severidade adequado com base no impacto nos negócios
- Forneça todas as informações relevantes no envio inicial do seu caso
- Inclua exemplos específicos e etapas de reprodução

- Anexe arquivos de log relevantes e capturas de tela de configuração
- Responda prontamente às solicitações de informações adicionais do engenheiro de suporte

Recursos adicionais de suporte:

- [AWS Gerenciamento de casos de suporte](#) - Guia para criar e gerenciar casos de suporte
- [AWS Planos de suporte](#) - Compare os recursos e os tempos de resposta do plano de suporte
- [Prepare-se para suportar cargas de trabalho](#) - orientação da Well-Architected para prontidão operacional
- [AWS Well-Architected](#) Framework - Melhores práticas para criar e operar cargas de trabalho em AWS

Segurança em AWS Elemental MediaTailor

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de data centers e arquiteturas de rede criados para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O [modelo de responsabilidade compartilhada](#) descreve isso como segurança da nuvem e segurança na nuvem:

- **Segurança da nuvem** — AWS é responsável por proteger a infraestrutura que executa AWS os serviços no Nuvem AWS. AWS também fornece serviços que você pode usar com segurança. Third-party auditores testam e verificam regularmente a eficácia de nossa segurança como parte dos Programas de Conformidade Programas de [AWS](#) de . Para saber mais sobre os programas de conformidade aplicáveis AWS Elemental MediaTailor, consulte [AWS Serviços no escopo do programa de conformidade AWS](#) .
- **Segurança na nuvem** — Sua responsabilidade é determinada pelo AWS serviço que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda você a entender como aplicar o modelo de responsabilidade compartilhada ao usar MediaTailor. Os tópicos a seguir mostram como configurar para atender MediaTailor aos seus objetivos de segurança e conformidade. Você também aprenderá a usar outros AWS serviços que ajudam a monitorar e proteger seus MediaTailor recursos.

Tópicos

- [Proteção de dados em AWS Elemental MediaTailor](#)
- [Identity and Access Management para AWS Elemental MediaTailor](#)
- [Validação de conformidade AWS Elemental MediaTailor](#)
- [Resiliência em AWS Elemental MediaTailor](#)
- [Segurança da infraestrutura em MediaTailor](#)
- [Cross-service prevenção delegada confusa](#)
- [Registro e monitoramento em MediaTailor](#)

Proteção de dados em AWS Elemental MediaTailor

O AWS [modelo de responsabilidade compartilhada](#) se aplica à proteção de dados no AWS Elemental MediaTailor. Conforme descrito neste modelo, AWS é responsável por proteger a infraestrutura global que executa todos os Nuvem AWS. Você é responsável por manter o controle sobre o conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos Serviços da AWS que usa. Para obter mais informações sobre privacidade de dados, consulte [Perguntas frequentes sobre privacidade de dados](#). Para obter informações sobre proteção de dados na Europa, consulte o [Centro de Regulamento Geral sobre a Proteção de Dados \(RGPD\)](#).

Para fins de proteção de dados, recomendamos que você proteja Conta da AWS as credenciais e configure usuários individuais com Centro de Identidade do AWS IAM ou AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com AWS os recursos. Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure a API e o registro de atividades do usuário com AWS CloudTrail. Para obter informações sobre o uso de CloudTrail trilhas para capturar AWS atividades, consulte Como [trabalhar com CloudTrail trilhas](#) no Guia AWS CloudTrail do usuário.
- Use soluções de AWS criptografia, juntamente com todos os controles de segurança padrão Serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sensíveis armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-3 ao acessar AWS por meio de uma interface de linha de comando ou de uma API, use um endpoint FIPS. Para saber mais sobre os endpoints FIPS disponíveis, consulte [Federal Information Processing Standard \(FIPS\) 140-3](#).

É altamente recomendável que nunca sejam colocadas informações confidenciais ou sensíveis, como endereços de e-mail de clientes, em tags ou campos de formato livre, como um campo Nome. Isso inclui quando você trabalha com MediaTailor ou Serviços da AWS usa o console, a API ou AWS os SDKs. AWS CLI Quaisquer dados inseridos em tags ou em campos de texto de formato livre

usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se você fornecer um URL para um servidor externo, é fortemente recomendável que não sejam incluídas informações de credenciais no URL para validar a solicitação nesse servidor.

Criptografia de dados

AWS Elemental MediaTailor não criptografa nem decifra dados em seu gerenciamento de manifestos de conteúdo ou em sua comunicação com servidores, CDNs ou players. MediaTailor não exige que você forneça dados do cliente ou outras informações confidenciais.

Não coloque informações confidenciais, como números de contas de clientes, informações de cartão de crédito ou credenciais de login, em campos de formato livre ou parâmetros de consulta. Isso se aplica a todos os usos do AWS Elemental MediaTailor, incluindo o console, a API, os SDKs e o AWS Command Line Interface (AWS CLI). Todos os dados inseridos por você no serviço podem ser separados para inclusão em logs de diagnóstico.

Ao fornecer um URL para um servidor externo, não inclua informações de credenciais não criptografadas no URL para validar a solicitação desse servidor.

Identity and Access Management para AWS Elemental MediaTailor

AWS Identity and Access Management (IAM) é uma ferramenta AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) a usar MediaTailor os recursos. O IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Tópicos

- [Público](#)
- [Autenticação com identidades](#)
- [Gerenciar o acesso usando políticas](#)
- [Como AWS Elemental MediaTailor funciona com o IAM](#)
- [Identity-based exemplos de políticas para AWS Elemental MediaTailor](#)
- [Resource-based exemplos de políticas para AWS Elemental MediaTailor](#)

- [AWS políticas gerenciadas para AWS Elemental MediaTailor](#)
- [Usando funções vinculadas a serviços para MediaTailor](#)
- [Solução de problemas AWS Elemental MediaTailor identidade e acesso](#)

Público

A forma como você usa AWS Identity and Access Management (IAM) difere com base na sua função:

- Usuário do serviço: solicite permissões ao seu administrador se você não conseguir acessar os atributos (consulte [Solução de problemas AWS Elemental MediaTailor identidade e acesso](#)).
- Administrador do serviço: determine o acesso do usuário e envie solicitações de permissão (consulte [Como AWS Elemental MediaTailor funciona com o IAM](#))
- Administrador do IAM: escreva políticas para gerenciar o acesso (consulte [Identity-based exemplos de políticas para AWS Elemental MediaTailor](#))

Autenticação com identidades

A autenticação é como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado como usuário do IAM ou assumindo uma função do IAM. Usuário raiz da conta da AWS

Você pode fazer login como uma identidade federada usando credenciais de uma fonte de identidade como Centro de Identidade do AWS IAM (IAM Identity Center), autenticação de login único ou credenciais. Google/Facebook Para ter mais informações sobre como fazer login, consulte [Como fazer login em sua Conta da AWS](#) no Guia do usuário do Início de Sessão da AWS .

Para acesso programático, AWS fornece um SDK e uma CLI para assinar solicitações criptograficamente. Para ter mais informações, consulte [AWS Signature Version 4 para solicitações de API](#) no Guia do usuário do IAM.

Conta da AWS usuário-raiz

Ao criar um Conta da AWS, você começa com uma identidade de login chamada usuário Conta da AWS raiz que tem acesso completo a todos Serviços da AWS os recursos. É altamente recomendável não usar o usuário-raiz em tarefas diárias. Consulte as tarefas que exigem credenciais de usuário-raiz em [Tarefas que exigem credenciais de usuário-raiz](#) no Guia do usuário do IAM.

Identidade federada

Como prática recomendada, exija que os usuários humanos usem a federação com um provedor de identidade para acessar Serviços da AWS usando credenciais temporárias.

Uma identidade federada é um usuário do seu diretório corporativo, provedor de identidade da web ou Directory Service que acessa Serviços da AWS usando credenciais de uma fonte de identidade. As identidades federadas assumem funções que oferecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, recomendamos Centro de Identidade do AWS IAM. Para saber mais, consulte [O que é o IAM Identity Center?](#) no Guia do usuário do Centro de Identidade do AWS IAM .

Usuários e grupos do IAM

Um [usuário do IAM](#) é uma identidade com permissões específicas para uma única pessoa ou aplicação. É recomendável usar credenciais temporárias, em vez de usuários do IAM com credenciais de longo prazo. Para obter mais informações, consulte [Exigir que usuários humanos usem a federação com um provedor de identidade para acessar AWS usando credenciais temporárias](#) no Guia do usuário do IAM.

Um [grupo do IAM](#) especifica um conjunto de usuários do IAM e facilita o gerenciamento de permissões para grandes conjuntos de usuários. Para ter mais informações, consulte [Casos de uso de usuários do IAM](#) no Guia do usuário do IAM.

Perfis do IAM

Uma [perfil do IAM](#) é uma identidade com permissões específicas que oferece credenciais temporárias. Você pode assumir uma função [mudando de um usuário para uma função do IAM \(console\)](#) ou chamando uma operação de AWS API AWS CLI ou. Para saber mais, consulte [Métodos para assumir um perfil](#) no Manual do usuário do IAM.

Os perfis do IAM são úteis para acesso de usuário federado, permissões de usuário do IAM temporárias, acesso entre contas, acesso entre serviços e aplicações em execução no Amazon EC2. Consulte mais informações em [Acesso a recursos entre contas no IAM](#) no Guia do usuário do IAM.

Gerenciar o acesso usando políticas

Você controla o acesso AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política define permissões quando associada a uma identidade ou recurso. AWS avalia essas

políticas quando um diretor faz uma solicitação. A maioria das políticas é armazenada AWS como documentos JSON. Para ter mais informações sobre documentos de política JSON, consulte [Visão geral das políticas JSON](#) no Guia do usuário do IAM.

Por meio de políticas, os administradores especificam quem tem acesso a que, definindo qual entidade principal pode realizar ações em quais recursos e sob quais condições.

Por padrão, usuários e perfis não têm permissões. Um administrador do IAM cria políticas do IAM e as adiciona aos perfis, os quais os usuários podem então assumir. As políticas do IAM definem permissões, independentemente do método usado para realizar a operação.

Identity-based políticas

Identity-based políticas são documentos de políticas de permissões JSON que você anexa a uma identidade (usuário, grupo ou função). Essas políticas controlam quais ações as identidades podem realizar, em quais recursos e sob quais condições. Para saber como criar uma política baseada em identidade, consulte [Definir permissões personalizadas do IAM com as políticas gerenciadas pelo cliente](#) no Guia do Usuário do IAM.

Identity-based as políticas podem ser políticas em linha (incorporadas diretamente em uma única identidade) ou políticas gerenciadas (políticas autônomas anexadas a várias identidades). Para saber como escolher entre uma política gerenciada e políticas em linha, consulte [Escolher entre políticas gerenciadas e políticas em linha](#) no Guia do usuário do IAM.

Resource-based políticas

Resource-based políticas são documentos de política JSON que você anexa a um recurso. Entre os exemplos estão políticas de confiança de perfil do IAM e políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. É necessário [especificar uma entidade principal](#) em uma política baseada em recursos.

Resource-based as políticas são políticas em linha localizadas nesse serviço. Você não pode usar políticas AWS gerenciadas do IAM em uma política baseada em recursos.

Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais que podem definir o máximo de permissões concedidas por tipos de políticas mais comuns:

- Limites de permissões: definem o número máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM. Para saber mais sobre limites de permissões, consulte [Limites de permissões para identidades do IAM](#) no Guia do usuário do IAM.
- Políticas de Controle de Serviços (SCPs): as SCPs especificam o número máximo de permissões para uma organização ou uma unidade organizacional no AWS Organizations. Para saber mais, consulte [Políticas de controle de serviço](#) no Guia do usuário do AWS Organizations .
- Políticas de controle de recursos (RCPs): definem o número máximo de permissões disponíveis para recursos em suas contas. Consulte mais informações em [Resource control policies \(RCPs\)](#) no Guia do usuário do AWS Organizations .
- Políticas de sessão: políticas avançadas transmitidas como um parâmetro durante a criação de uma sessão temporária para um perfil ou um usuário federado. Para saber mais, consulte [Políticas de sessão](#) no Guia do usuário do IAM.

Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como AWS determinar se uma solicitação deve ser permitida quando vários tipos de políticas estão envolvidos, consulte [Lógica de avaliação de políticas](#) no Guia do usuário do IAM.

Como AWS Elemental MediaTailor funciona com o IAM

Antes de usar o IAM para gerenciar o acesso MediaTailor, saiba com quais recursos do IAM estão disponíveis para uso MediaTailor.

Recursos do IAM que você pode usar com AWS Elemental MediaTailor

Recurso do IAM	MediaTailor apoio
Identity-based políticas	Sim
Resource-based políticas	Sim
Ações de políticas	Sim
Recursos de políticas	Não

Recurso do IAM	MediaTailor apoio
Chaves de condição de política (específicas do serviço)	Sim
ACLs	Não
ABAC (tags em políticas)	Parcial
Credenciais temporárias	Sim
Permissões de entidade principal	Sim
Perfis de serviço	Não
Service-linked funções	Sim

Para ter uma visão de alto nível de como MediaTailor e outros AWS serviços funcionam com a maioria dos recursos do IAM, consulte [AWS os serviços que funcionam com o IAM](#) no Guia do usuário do IAM.

Identity-based políticas para MediaTailor

Compatível com políticas baseadas em identidade: sim

Identity-based políticas são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como um usuário do IAM, um grupo de usuários ou uma função. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais atributos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Definir permissões personalizadas do IAM com as políticas gerenciadas pelo cliente](#) no Guia do Usuário do IAM.

Com as políticas baseadas em identidade do IAM, é possível especificar ações e recursos permitidos ou negados, assim como as condições sob as quais as ações são permitidas ou negadas. Para saber mais sobre todos os elementos que podem ser usados em uma política JSON, consulte [Referência de elemento de política JSON do IAM](#) no Guia do usuário do IAM.

Identity-based exemplos de políticas para MediaTailor

Para ver exemplos de políticas MediaTailor baseadas em identidade, consulte. [Identity-based exemplos de políticas para AWS Elemental MediaTailor](#)

Resource-based políticas dentro MediaTailor

Compatível com políticas baseadas em recursos: sim

O MediaTailor serviço oferece suporte a apenas um tipo de política baseada em recursos. É chamada de política de canal porque está vinculada a um canal. Essa política define quais diretores podem realizar ações no canal.

Resource-based políticas são documentos de política JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o atributo ao qual a política está anexada, a política define quais ações uma entidade principal especificado pode executar nesse atributo e em que condições. É necessário [especificar uma entidade principal](#) em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Para permitir o acesso entre contas, é possível especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em recursos. Consulte mais informações em [Acesso a recursos entre contas no IAM](#) no Guia do usuário do IAM.

Para saber como anexar uma política baseada em recursos a um canal, consulte. [Crie um canal usando o MediaTailor console](#)

Resource-based exemplos de políticas dentro MediaTailor

Para ver exemplos de políticas MediaTailor baseadas em recursos, consulte. [Resource-based exemplos de políticas para AWS Elemental MediaTailor](#)

Ações políticas para MediaTailor

Compatível com ações de políticas: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento `Action` de uma política JSON descreve as ações que podem ser usadas para permitir ou negar acesso em uma política. Incluem ações em uma política para conceder permissões para executar a operação associada.

Para ver uma lista de MediaTailor ações, consulte [Ações definidas por AWS Elemental MediaTailor](#) na Referência de Autorização de Serviço.

As ações de política MediaTailor usam o seguinte prefixo antes da ação:

```
mediatailor
```

Para especificar várias ações em uma única declaração, separe-as com vírgulas.

```
"Action": [  
    "mediatailor:action1",  
    "mediatailor:action2"  
]
```

Para ver exemplos de políticas MediaTailor baseadas em identidade, consulte. [Identity-based exemplos de políticas para AWS Elemental MediaTailor](#)

Recursos políticos para MediaTailor

Oferece compatibilidade com recursos de políticas: não

AWS Elemental MediaTailor não suporta a especificação de ARNs de recursos em uma política.

Chaves de condição de política para MediaTailor

Compatível com chaves de condição de política específicas de serviço: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento `Condition` especifica quando as instruções são executadas com base em critérios definidos. É possível criar expressões condicionais que usem [agentes de condição](#), como “igual a” ou “menor que”, para fazer a condição da política corresponder aos valores na solicitação. Para ver todas as chaves de condição AWS globais, consulte as [chaves de contexto de condição AWS global](#) no Guia do usuário do IAM.

Para obter uma lista de chaves de MediaTailor condição, consulte [Chaves de condição AWS Elemental MediaTailor](#) na Referência de autorização de serviço. Para saber com quais ações e

recursos você pode usar uma chave de condição, consulte [Ações definidas por AWS Elemental MediaTailor](#).

AWS Elemental MediaTailor não fornece chaves de condição específicas do serviço, mas oferece suporte ao uso de algumas chaves de condição globais. Para ver todas as chaves de condição AWS globais, consulte [Chaves de contexto de condição AWS global](#) no Guia AWS Identity and Access Management do usuário.

ACLs em MediaTailor

Compatível com ACLs: não

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou perfis da conta) têm permissões para acessar um recurso. As ACLs são semelhantes às políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

ABAC com MediaTailor

Compatível com ABAC (tags em políticas): parcial

Attribute-based controle de acesso (ABAC) é uma estratégia de autorização que define permissões com base em atributos chamados de tags. Você pode anexar tags a entidades e AWS recursos do IAM e, em seguida, criar políticas ABAC para permitir operações quando a tag do diretor corresponder à tag no recurso.

Para controlar o acesso baseado em tags, forneça informações sobre as tags no [elemento de condição](#) de uma política usando as `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` ou chaves de condição `aws:TagKeys`.

Se um serviço for compatível com as três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço for compatível com as três chaves de condição somente para alguns tipos de recursos, o valor será Parcial

Para saber mais sobre o ABAC, consulte [Definir permissões com autorização do ABAC](#) no Guia do usuário do IAM. Para visualizar um tutorial com etapas para configurar o ABAC, consulte [Usar controle de acesso por atributo \(ABAC\)](#) no Guia do usuário do IAM.

Para MediaTailor, use o valor Parcial.

Usando credenciais temporárias com MediaTailor

Compatível com credenciais temporárias: sim

As credenciais temporárias fornecem acesso de curto prazo aos AWS recursos e são criadas automaticamente quando você usa a federação ou troca de funções. AWS recomenda que você gere credenciais temporárias dinamicamente em vez de usar chaves de acesso de longo prazo. Para ter mais informações, consulte [Credenciais de segurança temporárias no IAM](#) e [Serviços da Serviços da AWS que funcionam com o IAM](#) no Guia do usuário do IAM.

Cross-service permissões principais para MediaTailor

Compatibilidade com o recurso de encaminhamento de sessões de acesso (FAS): sim

As sessões de acesso direto (FAS) usam as permissões do principal chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) de fazer solicitações aos serviços posteriores. Para obter detalhes da política ao fazer solicitações de FAS, consulte [Sessões de acesso direto](#).

Funções de serviço para MediaTailor

Compatível com perfis de serviço: não

AWS Elemental MediaTailor não suporta funções de serviço.

Service-linked funções para MediaTailor

Compatibilidade com perfis vinculados a serviços: sim

Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um. AWS service (Serviço da AWS) O serviço pode assumir a função de realizar uma ação em seu nome. Service-linked as funções aparecem no seu Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para perfis vinculados ao serviço.

Para obter detalhes sobre como criar ou gerenciar funções MediaTailor vinculadas a serviços, consulte. [Usando funções vinculadas a serviços para MediaTailor](#)

Identity-based exemplos de políticas para AWS Elemental MediaTailor

Por padrão, usuários e perfis não têm permissão para criar ou modificar recursos do MediaTailor. Para conceder permissão aos usuários para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM.

Para aprender a criar uma política baseada em identidade do IAM ao usar esses documentos de política em JSON de exemplo, consulte [Criar políticas do IAM \(console\)](#) no Guia do usuário do IAM.

Para obter detalhes sobre ações e tipos de recursos definidos por MediaTailor, incluindo o formato dos ARNs para cada um dos tipos de recursos, consulte [Ações, recursos e chaves de condição AWS Elemental MediaTailor na Referência de Autorização de Serviço](#).

Tópicos

- [Práticas recomendadas de política](#)
- [Usando o MediaTailor console](#)
- [Permitir que os usuários visualizem suas próprias permissões](#)

Práticas recomendadas de política

Identity-based as políticas determinam se alguém pode criar, acessar ou excluir MediaTailor recursos em sua conta. Essas ações podem incorrer em custos para sua Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas AWS gerenciadas e passe para as permissões de privilégios mínimos — Para começar a conceder permissões aos seus usuários e cargas de trabalho, use as políticas AWS gerenciadas que concedem permissões para muitos casos de uso comuns. Eles estão disponíveis no seu Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo AWS cliente que sejam específicas para seus casos de uso. Para saber mais, consulte [Políticas gerenciadas pela AWS](#) ou [Políticas gerenciadas pela AWS para funções de trabalho](#) no Guia do usuário do IAM.
- Aplique permissões de privilégio mínimo: ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em recursos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para saber mais sobre como usar o IAM para aplicar permissões, consulte [Políticas e permissões no IAM](#) no Guia do usuário do IAM.
- Use condições nas políticas do IAM para restringir ainda mais o acesso: é possível adicionar uma condição às políticas para limitar o acesso a ações e recursos. Por exemplo, é possível escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso às ações de serviço se elas forem usadas por meio de uma ação específica AWS service (Serviço da AWS), como CloudFormation. Para saber mais, consulte [Elementos da política JSON do IAM: condição](#) no Guia do usuário do IAM.

- Use o IAM Access Analyzer para validar suas políticas do IAM a fim de garantir permissões seguras e funcionais: o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam a linguagem de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de cem verificações de política e recomendações práticas para ajudar a criar políticas seguras e funcionais. Para saber mais, consulte [Validação de políticas do IAM Access Analyzer](#) no Guia do Usuário do IAM.
- Exigir autenticação multifator (MFA) — Se você tiver um cenário que exija usuários do IAM ou um usuário root, ative Conta da AWS a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para saber mais, consulte [Configuração de acesso à API protegido por MFA](#) no Guia do Usuário do IAM.

Para saber mais sobre as práticas recomendadas do IAM, consulte [Práticas recomendadas de segurança no IAM](#) no Guia do usuário do IAM.

Usando o MediaTailor console

Para acessar o AWS Elemental MediaTailor console, você deve ter um conjunto mínimo de permissões. Essas permissões devem permitir que você liste e visualize detalhes sobre os MediaTailor recursos em seu Conta da AWS. Caso crie uma política baseada em identidade mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para entidades (usuários ou perfis) com essa política.

Você não precisa permitir permissões mínimas do console para usuários que estão fazendo chamadas somente para a API AWS CLI ou para a AWS API. Em vez disso, permita o acesso somente a ações que correspondam à operação de API que estiverem tentando executar.

Para garantir que usuários e funções ainda possam usar o MediaTailor console, anexe também a política MediaTailor *ConsoleAccess* ou a política *ReadOnly* AWS gerenciada às entidades. Para obter informações, consulte [Adicionar permissões a um usuário](#) no Guia do usuário do IAM.

Permitir que os usuários visualizem suas próprias permissões

Este exemplo mostra como criar uma política que permita que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou programaticamente usando a API AWS CLI ou AWS .

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Sid": "ViewOwnUserInfo",
    "Effect": "Allow",
    "Action": [
      "iam:GetUserPolicy",
      "iam:ListGroupsForUser",
      "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": "*"
  }
]
```

Resource-based exemplos de políticas para AWS Elemental MediaTailor

Para saber como anexar uma política baseada em recursos a um canal, consulte. [Crie um canal usando o MediaTailor console](#)

Tópicos

- [Acesso anônimo](#)
- [Cross-account acesso](#)

Acesso anônimo

Considere a seguinte Allow política. Com essa política em vigor, MediaTailor permite acesso anônimo à `mediatailor:GetManifest` ação no recurso do canal na política. Isso ocorre onde **region** está o Região da AWS, **accountID** é seu Conta da AWS ID e **channelName** é o nome do recurso do canal.

Cross-account acesso

Considere a seguinte Allow política. Com essa política em vigor, MediaTailor permite a `mediatailor:GetManifest` ação no recurso do canal na política, em todas as contas. Isso ocorre onde **region** está o Região da AWS, **accountID** é seu Conta da AWS ID e **channelName** é o nome do recurso do canal.

AWS políticas gerenciadas para AWS Elemental MediaTailor

Para adicionar permissões a usuários, grupos e funções, é mais fácil usar políticas AWS gerenciadas do que escrever políticas você mesmo. É necessário tempo e experiência para criar [políticas gerenciadas pelo cliente do IAM](#) que fornecem à sua equipe apenas as permissões de que precisam. Para começar rapidamente, você pode usar nossas políticas AWS gerenciadas. Essas políticas abrangem casos de uso comuns e estão disponíveis na sua Conta da AWS. Para obter mais informações sobre políticas AWS gerenciadas, consulte [políticas AWS gerenciadas](#) no Guia do usuário do IAM.

AWS os serviços mantêm e atualizam as políticas AWS gerenciadas. Você não pode alterar as permissões nas políticas AWS gerenciadas. Os serviços ocasionalmente acrescentam permissões adicionais a uma política gerenciada pela AWS para oferecer suporte a novos recursos. Esse tipo de atualização afeta todas as identidades (usuários, grupos e funções) em que a política está anexada. É mais provável que os serviços atualizem uma política gerenciada pela AWS quando um novo recurso for iniciado ou novas operações se tornarem disponíveis. Os serviços não removem as permissões de uma política AWS gerenciada, portanto, as atualizações de políticas não violarão suas permissões existentes.

Além disso, AWS oferece suporte a políticas gerenciadas para funções de trabalho que abrangem vários serviços. Por exemplo, a política `ReadOnlyAccess` AWS gerenciada fornece acesso somente de leitura a todos os AWS serviços e recursos. Quando um serviço lança um novo recurso, AWS adiciona permissões somente de leitura para novas operações e recursos. Para obter uma lista e

descrições das políticas de perfis de trabalho, consulte [Políticas gerenciadas pela AWS para perfis de trabalho](#) no Guia do usuário do IAM.

AWS política gerenciada: AWSElementalMediaTailorFullAccess

É possível anexar a política AWSElementalMediaTailorFullAccess às suas identidades do IAM. É útil para usuários que precisam criar e gerenciar configurações de reprodução e recursos de montagem de canais, como programas e canais. Essa política concede permissões que permitem acesso total AWS Elemental MediaTailor a. Esses usuários podem criar, atualizar e excluir MediaTailor recursos.

Para visualizar as permissões para esta política, consulte [AWSElementalMediaTailorFullAccess](#) na Referência de políticas gerenciadas pela AWS .

AWS política gerenciada: AWSElementalMediaTailorReadOnly

É possível anexar a política AWSElementalMediaTailorReadOnly às suas identidades do IAM. É útil para usuários que precisam visualizar configurações de reprodução e recursos de montagem de canais, como programas e canais. Essa política concede permissões que permitem acesso somente para leitura a. AWS Elemental MediaTailor Esses usuários não podem criar, atualizar nem excluir recursos do MediaTailor :

Para visualizar as permissões para esta política, consulte [AWSElementalMediaTailorReadOnly](#) na Referência de políticas gerenciadas pela AWS .

MediaTailor atualizações para AWS políticas gerenciadas

Veja detalhes sobre as atualizações das políticas AWS gerenciadas MediaTailor desde que esse serviço começou a rastrear essas alterações. Para obter alertas automáticos sobre alterações feitas nesta página, inscreva-se no feed de RSS em MediaTailor [Histórico do documento para AWS Elemental MediaTailor](#).

Alteração	Descrição	Data
MediaTailor adicionou novas políticas gerenciadas	MediaTailor adicionou as seguintes políticas gerenciadas: AWSElementalMediaTailorReadOnly	24 de novembro de 2021

Alteração	Descrição	Data
	• AWS Elemental MediaTailor FullAccess	
MediaTailor começou a rastrear alterações	MediaTailor começou a rastrear as mudanças em suas políticas AWS gerenciadas.	24 de novembro de 2021

Usando funções vinculadas a serviços para MediaTailor

AWS Elemental MediaTailor usa funções [vinculadas ao serviço AWS Identity and Access Management](#) (IAM). Uma função vinculada ao serviço é um tipo exclusivo de função do IAM vinculada diretamente a. MediaTailor Service-linked as funções são predefinidas MediaTailor e incluem todas as permissões que o serviço exige para chamar outros AWS serviços em seu nome.

Uma função vinculada ao serviço facilita a configuração MediaTailor porque você não precisa adicionar manualmente as permissões necessárias. MediaTailor define as permissões de suas funções vinculadas ao serviço e, a menos que seja definido de outra forma, só MediaTailor pode assumir suas funções. As permissões definidas incluem a política de confiança e a política de permissões, que não pode ser anexada a nenhuma outra entidade do IAM.

Um perfil vinculado ao serviço poderá ser excluído somente após excluir seus atributos relacionados. Isso protege seus MediaTailor recursos porque você não pode remover inadvertidamente a permissão para acessar os recursos.

Para ter informações sobre outros serviços compatíveis com perfis vinculados ao serviço, consulte [Serviços da AWS que funcionam com o IAM](#) e procure os serviços que tiverem Sim na coluna Service-Linked Funções. Escolha um Sim com um link para visualizar a documentação do perfil vinculado para esse serviço.

Regiões suportadas para funções vinculadas a MediaTailor serviços

MediaTailor suporta o uso de funções vinculadas ao serviço em todas as regiões em que o serviço está disponível. Para mais informações, consulte [Regiões e endpoints da AWS](#).

Tópicos

- [Service-linked permissões de função para MediaTailor](#)

- [Criação de uma função vinculada ao serviço para MediaTailor](#)
- [Editando uma função vinculada ao serviço para MediaTailor](#)
- [Excluindo uma função vinculada ao serviço para MediaTailor](#)

Service-linked permissões de função para MediaTailor

MediaTailor usa a função vinculada ao serviço chamada `AWSServiceRoleForMediaTailor`—MediaTailor usa essa função vinculada ao serviço para invocar CloudWatch para criar e gerenciar grupos de log, fluxos de log e eventos de log. Essa função vinculada ao serviço é anexada à seguinte política gerenciada: `AWSMediaTailorServiceRolePolicy`.

A função `AWSServiceRoleForMediaTailor` vinculada ao serviço confia nos seguintes serviços para assumir a função:

- `mediatailor.amazonaws.com`

A política de permissões de função MediaTailor permite concluir as seguintes ações nos recursos especificados:

- Ação: `logs:PutLogEvents` em `arn:aws:logs:*:*:log-group:/aws/MediaTailor/*:log-stream:*`
- Ação: `logs:CreateLogStream`, `logs:CreateLogGroup`, `logs:DescribeLogGroups`, `logs:DescribeLogStreams` em `arn:aws:logs:*:*:log-group:/aws/MediaTailor/*`

É necessário configurar as permissões para permitir que uma entidade do IAM (como um usuário, grupo ou perfil) crie, edite ou exclua uma função vinculada ao serviço. Para obter mais informações, consulte [Permissões de Service-Linked função](#) no Guia do usuário do IAM.

Criação de uma função vinculada ao serviço para MediaTailor

Não é necessário criar manualmente um perfil vinculado ao serviço. Quando você ativa o registro de sessão no Console de gerenciamento da AWS, no AWS Command Line Interface (AWS CLI) ou na AWS API, MediaTailor cria a função vinculada ao serviço para você.

Important

Esse perfil vinculado ao serviço pode aparecer em sua conta se você concluiu uma ação em outro serviço que usa os atributos compatíveis com esse perfil. Além disso, se

Se você estava usando o MediaTailor serviço antes de 15 de setembro de 2021, quando ele começou a oferecer suporte a funções vinculadas ao serviço, MediaTailor criou a `AWSServiceRoleForMediaTailor` função em sua conta. Para saber mais, consulte [Uma Nova Função Apareceu na minha Conta do IAM](#).

Se excluir essa função vinculada ao serviço e precisar criá-la novamente, você pode usar esse mesmo processo para recriar a função na sua conta. Quando você ativa o registro de sessão, MediaTailor cria a função vinculada ao serviço para você novamente.

Você também pode usar o console do IAM para criar um perfil vinculado ao serviço com o caso de uso MediaTailor. Na AWS CLI ou na AWS API, crie uma função vinculada ao serviço com o nome `mediatailor.amazonaws.com` serviço. Para obter mais informações, consulte [Criação de uma Service-Linked função](#) no Guia do usuário do IAM. Se você excluir essa função vinculada ao serviço, será possível usar esse mesmo processo para criar a função novamente.

Editando uma função vinculada ao serviço para MediaTailor

MediaTailor não permite que você edite a função `AWSServiceRoleForMediaTailor` vinculada ao serviço. Depois que criar um perfil vinculado ao serviço, você não poderá alterar o nome do perfil, pois várias entidades podem fazer referência a ele. No entanto, será possível editar a descrição do perfil usando o IAM. Para obter mais informações, consulte [Edição de uma Service-Linked função](#) no Guia do usuário do IAM.

Excluindo uma função vinculada ao serviço para MediaTailor

Se você não precisar mais usar um atributo ou serviço que requer uma função vinculada a serviço, é recomendável excluí-la. Dessa forma, você não tem uma entidade não utilizada que não seja monitorada ativamente ou mantida. No entanto, você deve limpar os recursos de seu perfil vinculado ao serviço antes de excluí-lo manualmente.

Note

Se o MediaTailor serviço estiver usando a função quando você tentar limpar os recursos, a exclusão poderá falhar. Se isso acontecer, espere alguns minutos e tente a operação novamente.

Para limpar MediaTailor os recursos usados pelo `AWSServiceRoleForMediaTailor`

- Antes de excluir a função vinculada ao serviço criada por MediaTailor para a configuração de registro, você deve primeiro desativar todas as configurações de registro em sua conta. Para desativar uma configuração de log, defina o valor da porcentagem ativada como 0. Isso desativa todo o registro da sessão na configuração de reprodução correspondente. Para obter mais informações, consulte [Desativando uma configuração de log](#).

Como excluir manualmente o perfil vinculado a serviço usando o IAM

Use o console do IAM, o AWS Command Line Interface (AWS CLI) ou a AWS API para excluir a função `AWSServiceRoleForMediaTailor` vinculada ao serviço. Para obter mais informações, consulte [Excluir uma Service-Linked função](#) no Guia do usuário do IAM.

Solução de problemas AWS Elemental MediaTailor identidade e acesso

Use as informações a seguir para ajudá-lo a diagnosticar e corrigir problemas comuns que você pode encontrar ao trabalhar com MediaTailor um IAM.

Tópicos

- [Não estou autorizado a realizar uma ação em MediaTailor](#)
- [Não estou autorizado a realizar iam: PassRole](#)
- [Quero permitir que pessoas fora da minha Conta da AWS para acessar meus MediaTailor recursos](#)
- [MediaTailor não consigo acessar meu bucket Amazon S3](#)

Não estou autorizado a realizar uma ação em MediaTailor

Se você receber uma mensagem de erro informando que não tem autorização para executar uma ação, suas políticas deverão ser atualizadas para permitir que você realize a ação.

O erro do exemplo a seguir ocorre quando o usuário do IAM `mateojackson` tenta usar o console para visualizar detalhes sobre um atributo `my-example-widget` fictício, mas não tem as permissões `mediatailor:GetWidget` fictícias.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
mediatailor:GetWidget on resource: my-example-widget
```

Nesse caso, a política do usuário `mateojackson` deve ser atualizada para permitir o acesso ao recurso `my-example-widget` usando a ação `mediatailor:GetWidget`.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Não estou autorizado a realizar `iam:PassRole`

Se você receber uma mensagem de erro informando que não está autorizado a executar a ação `iam:PassRole`, as suas políticas devem ser atualizadas para permitir que você passe uma função para o MediaTailor.

Alguns Serviços da AWS permitem que você passe uma função existente para esse serviço em vez de criar uma nova função de serviço ou uma função vinculada ao serviço. Para fazê-lo, você deve ter permissões para passar o perfil para o serviço.

O exemplo de erro a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta utilizar o console para executar uma ação no MediaTailor. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Quero permitir que pessoas fora da minha Conta da AWS para acessar meus MediaTailor recursos

É possível criar um perfil que os usuários de outras contas ou pessoas fora da organização podem usar para acessar seus recursos. É possível especificar quem é confiável para assumir o perfil. Para serviços que oferecem compatibilidade com políticas baseadas em recursos ou listas de controle de acesso (ACLs), é possível usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se é MediaTailor compatível com esses recursos, consulte [Como AWS Elemental MediaTailor funciona com o IAM](#).
- Para saber como fornecer acesso aos seus recursos em todas as Contas da AWS que você possui, consulte Como [fornecer acesso a um usuário do IAM em outra Conta da AWS que você possui](#) no Guia do usuário do IAM.
- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte Como [fornecer acesso Contas da AWS a terceiros](#) no Guia do usuário do IAM.
- Para saber como conceder acesso por meio da federação de identidades, consulte [Conceder acesso a usuários autenticados externamente \(federação de identidades\)](#) no Guia do usuário do IAM.
- Para saber a diferença entre perfis e políticas baseadas em recurso para acesso entre contas, consulte [Acesso a recursos entre contas no IAM](#) no Guia do usuário do IAM.

MediaTailor não consegue acessar meu bucket Amazon S3

Se MediaTailor retornar erros ao tentar acessar o conteúdo do seu bucket do Amazon S3, verifique as seguintes causas comuns:

Problemas de permissões do Amazon S3

Sintomas de erro:

- 502 BadGatewayException com mensagens relacionadas ao Amazon S3
- 403 Forbidden erros nos MediaTailor registros
- Falha ao carregar conteúdo da origem do Amazon S3

Resolução:

1. Verifique se sua função MediaTailor de serviço tem as permissões necessárias do Amazon S3:
2. Verifique se sua política de bucket do Amazon S3 permite MediaTailor acesso a:

Problemas de acesso HTTPS

Sintomas de erro:

- SSL/TLS falhas no aperto de mão

- 502 Bad Gateway ao acessar URLs HTTPS do Amazon S3

Resolução:

1. Certifique-se de que sua política de bucket do Amazon S3 exija HTTPS:
2. Verifique se a MediaTailor configuração usa URLs HTTPS para conteúdo do Amazon S3.

Cross-account Acesso ao Amazon S3

Sintomas de erro:

- Access Denied erros quando o bucket do Amazon S3 está em uma conta diferente
- MediaTailor não pode assumir funções entre contas

Resolução:

1. Atualize a política de bucket do Amazon S3 na conta de conteúdo para confiar na função MediaTailor de serviço
2. Adicione permissões entre contas à função MediaTailor de serviço
3. Certifique-se de que ambas as contas tenham as relações de confiança adequadas configuradas

Validação de conformidade AWS Elemental MediaTailor

Para saber se um AWS service (Serviço da AWS) está dentro do escopo de programas de conformidade específicos, consulte [Serviços da AWS Escopo por Programa de Conformidade](#) [Serviços da AWS](#) e escolha o programa de conformidade em que você está interessado. Para obter informações gerais, consulte Programas de [AWS conformidade Programas AWS](#) de .

Você pode baixar relatórios de auditoria de terceiros usando AWS Artifact. Para obter mais informações, consulte [Baixar relatórios em AWS Artifact](#) .

Sua responsabilidade de conformidade ao usar Serviços da AWS é determinada pela confidencialidade de seus dados, pelos objetivos de conformidade de sua empresa e pelas leis e regulamentações aplicáveis. Para obter mais informações sobre sua responsabilidade de conformidade ao usar Serviços da AWS, consulte a [Documentação AWS de segurança](#).

Resiliência em AWS Elemental MediaTailor

A infraestrutura AWS global é construída em torno Regiões da AWS de zonas de disponibilidade. Regiões da AWS fornecem várias zonas de disponibilidade fisicamente separadas e isoladas, conectadas a redes de baixa latência, alta taxa de transferência e alta redundância. Com as zonas de disponibilidade, é possível projetar e operar aplicações e bancos de dados que automaticamente executam o failover entre as zonas sem interrupção. As zonas de disponibilidade são altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre zonas de disponibilidade Regiões da AWS e zonas de disponibilidade, consulte [Infraestrutura AWS global](#).

Além da infraestrutura AWS global, MediaTailor oferece vários recursos para ajudar a suportar suas necessidades de resiliência e backup de dados.

Segurança da infraestrutura em MediaTailor

Como serviço gerenciado, AWS Elemental MediaTailor é protegido pela segurança de rede AWS global. Para obter informações sobre serviços AWS de segurança e como AWS proteger a infraestrutura, consulte [AWS Cloud Security](#). Para projetar seu AWS ambiente usando as melhores práticas de segurança de infraestrutura, consulte [Proteção](#) de infraestrutura no Security Pillar AWS Well-Architected Framework.

Você usa chamadas de API AWS publicadas para acessar MediaTailor pela rede. Os clientes devem oferecer compatibilidade com:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Suítes de criptografia com sigilo direto perfeito (PFS), como DHE (Ephemeral) ou ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Diffie-Hellman A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

Você pode chamar essas operações de API de qualquer local de rede, mas MediaTailor oferece suporte a políticas de acesso baseadas em recursos, que podem incluir restrições com base no endereço IP de origem. Você também pode usar MediaTailor políticas para controlar o acesso de endpoints específicos da Amazon Virtual Private Cloud (Amazon VPC) ou VPCs específicas. Efetivamente, isso isola o acesso à rede a um determinado MediaTailor recurso somente da VPC específica dentro da AWS rede.

Cross-service prevenção delegada confusa

O problema "confused deputy" é um problema de segurança em que uma entidade que não tem permissão para executar uma ação pode coagir uma entidade mais privilegiada a executar a ação. Em AWS, a falsificação de identidade entre serviços pode resultar no problema confuso do deputado. Cross-service a representação pode ocorrer quando um serviço (o serviço de chamada) chama outro serviço (o serviço chamado). O serviço de chamada pode ser manipulado de modo a usar suas permissões para atuar nos recursos de outro cliente de uma forma na qual ele não deveria ter permissão para acessar. Para evitar isso, a AWS fornece ferramentas que ajudam você a proteger seus dados para todos os serviços com entidades principais de serviço que receberam acesso aos recursos em sua conta.

Recomendamos usar as chaves de contexto de condição SourceAccount global [aws: SourceArn](#) e [aws:](#) nas políticas de recursos para limitar as permissões que AWS Elemental MediaTailor concedem outro serviço ao recurso. Se você utilizar ambas as chaves de contexto de condição global, o valor `aws:SourceAccount` e a conta `aws:SourceArn` no valor deverão utilizar o mesmo ID de conta quando utilizados na mesma instrução de política.

O valor de `aws:SourceArn` deve ser a configuração de reprodução que MediaTailor publica CloudWatch registros em sua região e conta. No entanto, isso só se aplica se você usar a [MediaTailorLogger](#) função que permite MediaTailor publicar CloudWatch registros da Amazon em sua conta. Isso não se aplica se você usar uma [função vinculada ao serviço](#) para permitir a MediaTailor publicação dos CloudWatch registros.

A maneira mais eficaz de se proteger do problema 'confused deputy' é usar a chave de contexto de condição global `aws:SourceArn` com o ARN completo do recurso. Se você não souber o ARN completo do recurso ou se especificar vários recursos, use a chave de condição de contexto global `aws:SourceArn` com curingas (*) para as partes desconhecidas do ARN. Por exemplo, `.arn:aws:service::123456789012:*`

O exemplo a seguir mostra como você pode usar as chaves de contexto de condição `aws:SourceAccount` global `aws:SourceArn` e as chaves de contexto MediaTailor para evitar o confuso problema substituto.

Registro e monitoramento em MediaTailor

Esta seção fornece uma visão geral das opções de registro em log e monitoramento no AWS Elemental MediaTailor para fins de segurança. Para obter mais informações sobre registro e

monitoramento, MediaTailor consulte [Monitorar e atribuir tags \(tagging\) AWS Elemental MediaTailor recursos](#).

O monitoramento é uma parte importante da manutenção da confiabilidade, disponibilidade e desempenho de AWS Elemental MediaTailor suas AWS soluções. Você deve coletar dados de monitoramento de todas as partes da sua AWS solução para poder depurar com mais facilidade uma falha de vários pontos, caso ocorra. AWS fornece várias ferramentas para monitorar seus MediaTailor recursos e responder a possíveis incidentes:

CloudWatch Alarmes da Amazon

Usando CloudWatch alarmes, você observa uma única métrica durante um período de tempo especificado por você. Se a métrica exceder um determinado limite, uma notificação será enviada para um tópico do Amazon SNS ou para uma política de Auto AWS Scaling. CloudWatch os alarmes não invocam ações porque estão em um estado específico. O estado deve ter sido alterado e mantido por uma quantidade especificada de períodos. Para obter mais informações, consulte [the section called “Monitoramento com CloudWatch métricas”](#).

AWS CloudTrail logs

CloudTrail fornece um registro das ações realizadas por um usuário, função ou AWS serviço em AWS Elemental MediaTailor. Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita MediaTailor, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais. Para obter mais informações, consulte [Gravação AWS Elemental MediaTailor Chamadas de API](#).

AWS Trusted Advisor

Trusted Advisor baseia-se nas melhores práticas aprendidas ao atender centenas de milhares de AWS clientes. Trusted Advisor inspeciona seu AWS ambiente e, em seguida, faz recomendações quando existem oportunidades para economizar dinheiro, melhorar a disponibilidade e o desempenho do sistema ou ajudar a fechar lacunas de segurança. Todos os AWS clientes têm acesso a cinco verificações do Trusted Advisor. Os clientes com um plano de suporte Business ou Enterprise podem ver todas as verificações do Trusted Advisor .

Para obter mais informações, consulte [AWS Trusted Advisor](#).

Monitorar e atribuir tags (tagging) AWS Elemental MediaTailor recursos

O monitoramento é uma parte importante da manutenção da confiabilidade, disponibilidade e desempenho de AWS Elemental MediaTailor suas outras AWS soluções. AWS fornece as seguintes ferramentas de monitoramento para observar MediaTailor, relatar quando algo está errado e realizar ações automáticas quando apropriado:

- A Amazon CloudWatch monitora seus AWS recursos e os aplicativos nos quais você executa AWS em tempo real. Você pode coletar e rastrear métricas, criar painéis personalizados e definir alarmes que o notificam ou que realizam ações quando uma métrica especificada atinge um limite definido. Por exemplo, você pode CloudWatch rastrear o uso da CPU ou outras métricas de suas instâncias do Amazon EC2 e iniciar automaticamente novas instâncias quando necessário. Para obter mais informações, consulte o [Guia CloudWatch do usuário da Amazon](#).
- O Amazon CloudWatch Logs permite que você monitore, armazene e acesse seus arquivos de log de todas as interações com seu servidor de decisão de anúncios (ADS). AWS Elemental MediaTailor emite registros de solicitações de anúncios, redirecionamentos, respostas e relatórios de solicitações e respostas. Erros do ADS e dos servidores de origem também são emitidos para grupos de registros na Amazon CloudWatch. MediaTailor também fornece informações sobre anúncios ignorados e os motivos pelos quais eles foram ignorados. Para obter mais informações, consulte [Solução de problemas com o ignorar anúncios](#). É possível também arquivar seus dados de log em armazenamento resiliente. Para obter informações gerais, consulte o [Guia do usuário do Amazon CloudWatch Logs](#). Para obter informações sobre os registros do ADS e como acessá-los para análise por meio do Amazon CloudWatch Logs Insights, consulte [AWS Elemental MediaTailor Análise de registros do ADS no Amazon CloudWatch Logs Insights](#).

Tópicos

- [Visualizar AWS Elemental MediaTailor logs](#)
- [Monitoramento AWS Elemental MediaTailor com CloudWatch métricas da Amazon](#)
- [Gravação AWS Elemental MediaTailor Chamadas de API](#)
- [Recebendo AWS Elemental MediaTailor alertas de montagem de canais](#)
- [Tags AWS Elemental MediaTailor recursos](#)
- [Monitoramento AWS serviços de mídia com monitor de fluxo de trabalho](#)

Visualizar AWS Elemental MediaTailor logs

MediaTailor emite registros que descrevem uma variedade de marcos e atividades em canais e configurações de reprodução. Você pode usar esses registros para obter visibilidade do seu fluxo de trabalho e solucionar problemas com o serviço. Os tópicos a seguir descrevem registros e opções de registro.

Tópicos

- [AWS Elemental MediaTailor Descrição dos registros do ADS e tipos de eventos](#)
- [AWS Elemental MediaTailor descrição dos registros de manifesto e tipos de eventos](#)
- [AWS Elemental MediaTailor transcodificar registros, descrição e tipos de eventos](#)
- [Usando registros vendidos para enviar AWS Elemental MediaTailor logs](#)
- [Writing AWS Elemental MediaTailor registra diretamente no Amazon CloudWatch Logs](#)
- [Controlando o volume de AWS Elemental MediaTailor logs](#)
- [Filtrar AWS Elemental MediaTailor registros e eventos](#)
- [Generating \(Gerando\) AWS Elemental MediaTailor registros de depuração](#)

AWS Elemental MediaTailor Descrição dos registros do ADS e tipos de eventos

As seções a seguir descrevem os registros que são MediaTailor emitidos para descrever eventos com o servidor de decisão de anúncios (ADS). São AdDecisionServerInteractions registros.

Tópicos

- [AdDecisionServerInteractions eventos](#)
- [Descrição do registro do ADS](#)
- [Esquema JSON de log do ADS](#)

AdDecisionServerInteractions eventos

Os eventos a seguir são emitidos durante MediaTailor as interações com o servidor de decisão de anúncios (ADS).

Log	Description
AD_MARKER_FOUND	MediaTailor encontrou um marcador de anúncio no manifesto.
BEACON_FIRED	Um sinal de rastreamento foi acionado para relatar eventos publicitários. No modo de relatório do lado do servidor (padrão), MediaTailor dispara o beacon. No modo de relatório do lado do cliente, o dispositivo de reprodução dispara o farol.
EMPTY_VAST_RESPONSE	O ADS retornou uma resposta VAST vazia contendo zero anúncios.
EMPTY_VMAP_RESPONSE	O ADS retornou uma resposta VMAP vazia.
ERROR_ADS_INVALID_RESPONSE	O ADS retornou um código de status diferente de 200.
ERROR_ADS_IO	MediaTailor encontrou um erro ao tentar se comunicar com o ADS.
ERROR_ADS_RESPONSE_PARSE	MediaTailor encontrou um erro ao analisar a resposta do ADS.
ERROR_ADS_RESPONSE_UNKNOWN_ROOT_ELEMENT	A resposta do ADS contém um elemento raiz inválido.
ERROR_ADS_TIMEOUT	A MediaTailor solicitação para o ADS atingiu o tempo limite.
ERROR_DISALLOWED_HOST	O host do ADS não é permitido.
ERROR_FIRING_BEACON_FAILED	MediaTailor falhou em disparar o farol de rastreamento.
ERROR_PERSONALIZATION_DISABLED	A inserção de anúncios está desativada nesta sessão.

Log	Description
ERROR_UNKNOWN	MediaTailor encontrou um erro desconhecido durante a solicitação do ADS.
ERROR_UNKNOWN_HOST	O host do ADS é desconhecido.
ERROR_VAST_INVALID_MEDIA_FILE	O VAST Ad tem um MediaFile elemento inválido ou ausente.
ERROR_VAST_INVALID_VAST_AD_TAG_URI	A resposta VAST contém uma mensagem inválidaVASTAdTagURI .
ERROR_VAST_MISSING_CREATIVES	O Ad VAST contém zero ou vários Creatives elementos. É necessário exatamente um Creatives elemento.
ERROR_VAST_MISSING_IMPRESSION	O Ad VAST contém zero Impression elementos. É necessário pelo menos um Impression elemento.
ERROR_VAST_MISSING_MEDIAFILES	O Ad VAST contém zero ou vários MediaFiles elementos. É necessário exatamente um MediaFiles elemento.
ERROR_VAST_MISSING_OVERLAYS	MediaTailor não recebeu nenhum criativo não linear do servidor de anúncios.
ERROR_VAST_MULTIPLE_LINEAR	O Ad VAST contém vários Linear elementos.
ERROR_VAST_MULTIPLE_TRACKING_EVENTS	O Ad VAST contém vários TrackingEvents elementos.
ERROR_VAST_REDIRECT_EMPTY_RESPONSE	A solicitação de redirecionamento VAST retornou uma resposta vazia.
ERROR_VAST_REDIRECT_FAILED	A solicitação de redirecionamento VAST encontrou um erro.

Log	Description
ERROR_VAST_REDIRECT_MULTIPLE_VAST	A solicitação de redirecionamento VAST retornou vários anúncios.
FILLED_AVAIL	MediaTailor preencheu com sucesso a disponibilidade.
FILLED_OVERLAY_AVAIL	MediaTailor preencheu com sucesso a sobreposição disponível.
INTERSTITIAL_VOD_FAILURE	A solicitação ou resposta do ADS encontrou um problema ao preencher os recursos intersticiais da playlist VOD. Nenhum anúncio será inserido.
INTERSTITIAL_VOD_SUCCESS	MediaTailor disponibilidades intersticiais preenchidas com sucesso para a playlist VOD.
MAKING_ADS_REQUEST	MediaTailor está solicitando anúncios do ADS.
MODIFIED_TARGET_URL	MediaTailor modificou o URL de destino de saída.
NON_AD_MARKER_FOUND	MediaTailor encontrou um marcador de anúncio não acionável no manifesto.
PRE_ADS_REQUEST_FUNCTION_COMPLETED	Uma função individual no gancho de solicitação de pré-anúncios foi concluída. Esse é um tipo de evento opcional.
PRE_ADS_REQUEST_FUNCTION_ERROR	Uma função individual no gancho de solicitação de pré-anúncios falhou.
PRE_ADS_REQUEST_HOOK_ERROR	A execução do gancho de solicitação de pré-anúncios falhou.

Log	Description
PRE_ADS_REQUEST_HOOK_SUMMARY	Resumo da execução do gancho de solicitações pré-anúncios, incluindo status de sucesso ou erro. Esse é um tipo de evento opcional.
RAW_ADS_RESPONSE	MediaTailor recebeu uma resposta bruta do ADS.
REDIRECTED_VAST_RESPONSE	MediaTailor recebeu uma resposta VAST após seguir o redirecionamento VAST.
VAST_REDIRECT	A resposta do anúncio VAST contém um redirecionamento.
VAST_RESPONSE	MediaTailor recebeu uma resposta VAST.
VOD_TIME_BASED_AVAIL_PLAN_SUCCESS	MediaTailor criou com sucesso um plano de disponibilidade baseado em tempo para o modelo VOD.
VOD_TIME_BASED_AVAIL_PLAN_VAST_RESPONSE_FOR_OFFSET	MediaTailor está criando um plano de disponibilidade baseado em tempo para o modelo de VOD. MediaTailor recebeu uma resposta VAST para o intervalo de tempo.
VOD_TIME_BASED_AVAIL_PLAN_WARNING_NO_ADVERTISEMENTS	A solicitação ou resposta do ADS encontrou um problema ao criar um plano de disponibilidade baseado em tempo para o modelo de VOD. Nenhum anúncio será inserido.
WARNING_NO_ADVERTISEMENTS	MediaTailor encontrou um problema ao preencher o formulário. Nenhum anúncio é inserido.
WARNING_URL_VARIABLE_SUBSTITUTION_FAILED	MediaTailor não pode substituir variáveis dinâmicas no URL do ADS. Verifique a configuração do URL.

Log	Description
WARNING_VPAID_AD_DROPPED	Um anúncio VPAID foi descartado devido à falta de uma lista, ou a sessão usa relatórios do lado do servidor.

Descrição do registro do ADS

Esta seção descreve a estrutura e o conteúdo da descrição de logs do ADS. Para explorar por conta própria em um editor JSON, use a listagem em [the section called “Esquema JSON de log do ADS”](#).

Cada evento no registro do ADS contém os campos padrão que são gerados pelos CloudWatch registros. Para obter mais informações, consulte [Analisar dados de registro com o CloudWatch Logs Insights](#).

Propriedades de registros do ADS

Esta seção descreve as propriedades dos logs do ADS.

Propriedade	Tipo	Obrigatório	Descrição
adsRequestUrl	string	false	O URL completo da solicitação de ADS feita por MediaTailor.
avail	objeto do tipo disponibilidade	false	Informações sobre um item que está MediaTailor cheio de anúncios. Atualmente, para o tipo de FILLED_AVAIL evento, esse é o plano criado MediaTailor quando ele encontra a disponibilidade pela primeira vez. A forma como a disponibi

Propriedade	Tipo	Obrigatório	Descrição
			lidade é, por fim, preenchida pode variar desse plano, dependendo de como o conteúdo é reproduzido.
awsAccountId	string	true	O ID da conta da AWS para a MediaTailor configuração que foi usada para a sessão.
customerId	string	true	A versão com hash do ID da conta da AWS, que é possível usar para correlacionar várias entradas de log.
eventDescription	string	true	Uma breve descrição do evento que acionou essa mensagem de log, fornecida pelo MediaTailor serviço. Por padrão, permanece em branco. Exemplo: Got VAST response.
eventTimestamp	string	true	A data e a hora do evento.

Propriedade	Tipo	Obrigatório	Descrição
eventType	string	true	O código do evento que acionou essa mensagem de log. Exemplo: VAST_RESPONSE .
errorType	string	false	Classificação estruturada de erros. Presente em PRE_ADS_REQUEST_FUNCTION_ERROR eventos PRE_ADS_REQUEST_H0_OK_ERROR e eventos. Valores:INTERNAL_ERROR ,SYNTAX_ERROR ,RESOURCE_LIMIT_ERROR ,RESTRICTION_ERROR ,TIMEOUT_ERROR ,VALIDATION_ERROR .

Propriedade	Tipo	Obrigatório	Descrição
cause	string	false	Cadeia de detalhes do erro descrevendo a falha específica. Presente em PRE_ADS_REQUEST_ERROR eventos PRE_ADS_REQUEST_ERROR e eventos.
eventId	string	false	Identificador exclusivo por invocação de gancho. Use esse campo para correlacionar eventos em nível de gancho e em nível de função para a mesma execução. Presente somente em eventos do Functions.
executionTimeMs	integer	false	Tempo de execução em milissegundos. Presente somente em eventos do Functions.

Propriedade	Tipo	Obrigatório	Descrição
<code>functionId</code>	string	false	O identificador da função. Em eventos de gancho, essa é a função configurada no gancho. Em eventos de função, essa é a função individual dentro do executor. Presente somente em eventos do Functions.
<code>functionType</code>	string	false	O tipo de função: <code>CUSTOM_OUTPUT</code> , <code>HTTP_REQUEST</code> , <code>SEQUENTIAL_EXECUTOR</code> , ou <code>CONCURRENT_EXECUTOR</code> . Presente somente em eventos de nível de função.

Propriedade	Tipo	Obrigatório	Descrição
http	objeto	false	Detalhes da solicitação e resposta HTTP. Presente em eventos de nível de função somente para HTTP_REQUEST funções. Contém request (url, método, cabeçalhos, corpo), responseTimeMs , response (statusCode, cabeçalhos, corpo) truncated e (verdadeiro quando o corpo da solicitação ou da resposta foi truncado para caber nos limites de tamanho do evento de log).
input	objeto	false	Before-values de variáveis direcionadas por expressões de saída. Presente somente em eventos de nível de função.

Propriedade	Tipo	Obrigatório	Descrição
output	objeto	false	Valores produzidos pela função. Presente somente em PRE_ADS_REQUEST_COMPLETED eventos.
status	string	false	Resultado da execução do gancho: SUCCESS ou ERROR. Presente somente em PRE_ADS_REQUEST_HOOK_SUMMARY eventos.
originId	string	true	O nome da MediaTailor configuração. Isso é diferente da fonte de conteúdo de vídeo, que também faz parte da configuração.
prefetchScheduleName	string	false	O nome da agenda de pré-busca associada a esse evento publicitário.

Propriedade	Tipo	Obrigatório	Descrição
<code>requestHeaders</code>	matriz do tipo cabeçalhos de solicitações	false	Os cabeçalhos MediaTailor incluídos na solicitação do ADS. Normalmente, os logs os incluem quando ocorre uma falha em uma solicitação ao ADS, para ajudar na solução de problemas.
<code>requestId</code>	string	true	O ID da MediaTailor solicitação, que você pode usar para correlacionar várias entradas de registro para a mesma solicitação.
<code>sessionId</code>	string	true	O identificador numérico exclusivo MediaTailor atribuído à sessão do jogador. Todas as solicitações que um player faz para uma sessão têm o mesmo ID de sessão. Exemplo: e039fd39-09f0-46b2-aca9-9871cc116cde .
<code>sessionType</code>	string (valores legais: [DASH, HLS])	true	O tipo de fluxo do player.

Propriedade	Tipo	Obrigatório	Descrição
vastAd	objeto do tipo vastAd	false	Informações sobre um único anúncio analisado na resposta VAST.
vastResponse	objeto do tipo vastResponse	false	Informações sobre a resposta VAST MediaTailor recebida do ADS.
vodCreativeOffsets	objeto do tipo vodCreativeOffsets	false	Um mapa que indica os deslocamentos de tempo no manifesto onde MediaTailor serão inseridos os resultados, com base na resposta do VMAP.
vodVastResponseTimeOffset	número	false	O deslocamento de tempo específico do VMAP para inserção de anúncios VOD.

adContent

Esta seção descreve as propriedades de adContent dos logs do ADS.

Propriedades de adContent dos logs do ADS

Propriedade	Tipo	Obrigatório	Description
adPlaylistUri	objeto do tipo anúncio PlaylistUri	false	O mapeamento do manifesto de origem de uma variante para o manifesto de

Propriedade	Tipo	Obrigatório	Description
			anúncio da variante. Para DASH, ele contém uma única entrada, pois todas as variantes são representadas em um único manifesto DASH.

anúncio PlaylistUri

Esta seção descreve as propriedades do anúncio de registros do ADSPlaylistUri.

Registros e PlaylistUri propriedades do ADS

Propriedade	Tipo	Obrigatório	Descrição
<any string>	string	false	O URL do manifesto de anúncio para a variante específica.

disponibilidade

Esta seção descreve as propriedades de disponibilidade dos logs do ADS.

Propriedades de disponibilidade dos logs do ADS

Propriedade	Tipo	Obrigatório	Descrição
availId	string	true	O identificador exclusivo para essa disponibilidade. Para HLS, este é o número de sequência de mídia em que a disponibilidade começa. Para DASH,

Propriedade	Tipo	Obrigatório	Descrição
			esse é o ID do período.
creativeAds	matriz do tipo creativeAd	true	Os anúncios MediaTailor inseridos no avail.
fillRate	número	true	A taxa na qual os anúncios preenchem a duração da disponibilidade, de 0,0 (para 0%) a 1,0 (para 100%).
filledDuration	número	true	A soma das durações de todos os anúncios inseridos na disponibilidade.
numAds	número	true	O número de anúncios MediaTailor inseridos no avail.
originAvailabilityDuration	número	true	A duração da disponibilidade conforme especificado no streaming de conteúdo da origem (CUE_OUT ou SCTE).

Propriedade	Tipo	Obrigatório	Descrição
skippedAds	matriz do tipo skippedAd	false	Os anúncios que MediaTailor não foram inseridos , por motivos como TRANSCODE_IN_PROGRESS TRANSCODE_ERROR e. Para obter uma lista dos motivos pelos quais o anúncio foi ignorado, consulte. Solução de problemas com o ignorar anúncios
slateAd	objeto do tipo slateAd	true	Informações sobre o anúncio em ardósia, MediaTailor usado para preencher todos os segmentos não preenchidos na disponibilidade.

creativeAd

Esta seção descreve as propriedades de creativeAd dos logs do ADS.

Propriedades de creativeAd dos logs do ADS

Propriedade	Tipo	Obrigatório	Description
adContent	objeto do tipo adContent	true	Informações sobre o conteúdo do anúncio inserido.

Propriedade	Tipo	Obrigatório	Description
<code>creativeUniqueId</code>	string	true	O identificador exclusivo do anúncio, usado como uma chave para transcodificação. Este é o campo de ID do criador na resposta VAST, se disponível. Caso contrário, será o URL de mezanino do anúncio.
<code>trackingEvents</code>	objeto do tipo trackingEvents	true	Os URLs de beacon de rastreamento para os vários eventos de rastreamento do anúncio. As chaves são os nomes de eventos, e os valores são uma lista de URLs de beacon.
<code>transcodedAdDuration</code>	número	true	A duração do anúncio, calculada a partir do ativo transcodificado.
<code>uri</code>	string	true	O URL da versão de mezanino do anúncio, que é a entrada para o transcodificador.

Propriedade	Tipo	Obrigatório	Description
vastDuration	número	true	A duração do anúncio, conforme analisada na resposta VAST.

cabeçalhos de solicitações

Esta seção descreve as propriedades dos cabeçalhos de solicitações de logs do ADS.

Propriedades dos cabeçalhos de solicitações de logs do ADS

Propriedade	Tipo	Obrigatório	Descrição
name	string	true	O nome do cabeçalho.
value	string	true	O valor do cabeçalho.

skippedAd

Esta seção descreve as propriedades de skippedAd dos logs do ADS.

Propriedades de skippedAd dos logs do ADS

Propriedade	Tipo	Obrigatório	Descrição
adMezzanineUrl	string	true	O URL de mezanino do anúncio ignorado.
creativeUniqueId	string	true	O identificador exclusivo do anúncio, usado como uma chave para transcodificação. Este é o campo de ID do criador na resposta VAST, se disponível.

Propriedade	Tipo	Obrigatório	Descrição
			Caso contrário, será o URL de mezanino do anúncio.
skippedReason	string	true	O código que indica por que o anúncio não foi inserido. Exemplo: TRANSCODE_IN_PROGRESS . Para obter uma lista dos motivos pelos quais o anúncio foi ignorado, consulte. Solução de problemas com o ignorar anúncios
transcodeAdDuration	número	false	A duração do anúncio, calculada a partir do ativo transcodificado.
vastDuration	número	true	A duração do anúncio, conforme analisada na resposta VAST.

slateAd

Esta seção descreve as propriedades de slateAd dos logs do ADS.

Propriedades de slateAd dos logs do ADS

Propriedade	Tipo	Obrigatório	Description
adContent	objeto do tipo adContent	true	Informações sobre o conteúdo do anúncio inserido.
creativeUniqueId	string	true	O identificador exclusivo do anúncio, usado como uma chave para transcodificação. Este é o campo de ID do criador na resposta VAST, se disponível. Caso contrário, será o URL de mezanino do anúncio.
transcodeAdDuration	número	true	A duração do anúncio, calculada a partir do ativo transcodificado.
uri	string	true	O URL da versão de mezanino do anúncio, que é a entrada para o transcodificador.

trackingEvents

Esta seção descreve as propriedades de trackingEvents dos logs do ADS.

Propriedades de trackingEvents dos logs do ADS

Propriedade	Tipo	Obrigatório	Description
<any string>	matriz do tipo string	false	A lista de URLs de beacon para o evento de rastreamento especificado (impressão, conclusão, etc.)

vastAd

Esta seção descreve as propriedades de vastAd dos logs do ADS.

Propriedades de vastAd dos logs do ADS

Propriedade	Tipo	Obrigatório	Descrição
adSystem	string	true	O valor da tag AdSystem na resposta VAST.
adTitle	string	true	Os arquivos de mídia disponíveis para o anúncio na resposta VAST.
creativeAdId	string	true	O valor do atributo adId da tag Creative na resposta VAST.
creativeId	string	true	O valor do atributo id da tag Creative na resposta VAST.
duration	número	true	A duração aproximada do anúncio,

Propriedade	Tipo	Obrigatório	Descrição
			com base na tag <code>duration</code> no elemento <code>linear</code> da resposta VAST.
<code>trackingEvents</code>	objeto do tipo trackingEvents	true	Os URLs de beacon de rastreamento para os vários eventos de rastreamento do anúncio. As chaves são os nomes de eventos, e os valores são uma lista de URLs de beacon.
<code>vastAdId</code>	string	true	O valor do atributo <code>id</code> da tag <code>Ad</code> na resposta VAST
<code>vastAdTagUri</code>	string	false	O URI de VMAP-specific redirecionamento de um anúncio.
<code>vastMediaFiles</code>	matriz do tipo vastoMediaFile	true	A lista de arquivos de mídia disponíveis para o anúncio na resposta VAST.

vasto MediaFile

Esta seção descreve as propriedades dos vastos registros do ADSMediaFile.

O ADS registra MediaFile vastas propriedades

Propriedade	Tipo	Obrigatório	Descrição
apiFramework	string	true	A estrutura de API necessária para gerenciar o arquivo de mídia. Exemplo: VPAID.
bitrate	número	true	A taxa de bits do arquivo de mídia.
delivery	string	true	O protocolo usado para o arquivo de mídia, definido como progressivo ou streaming.
height	número	true	A altura em pixels do arquivo de mídia.
id	string	true	O valor do atributo id da tag MediaFile .
type	string	true	O tipo MIME do arquivo de mídia, obtido do atributo type da tag MediaFile .
uri	string	true	O URL da versão de mezanino do anúncio, que é a entrada para o transcodificador.
width	número	true	A largura em pixels do arquivo de mídia.

vastResponse

Esta seção descreve as propriedades de vastResponse dos logs do ADS.

Propriedades de vastResponse dos logs do ADS

Propriedade	Tipo	Obrigatório	Description
errors	matriz do tipo string	true	Os URLs de erro analisados nas tags Error na resposta VAST.
vastAds	matriz do tipo vastAd	true	Os anúncios analisados na resposta VAST.
version	string	true	A versão da especificação VAST, analisada no atributo version da tag VAST na resposta.

vod CreativeOffsets

Esta seção descreve as propriedades dos registros do ADS vodCreativeOffsets.

Registros e propriedades do ADS CreativeOffsets

Propriedade	Tipo	Obrigatório	Description
<any string>	matriz do tipo vod CreativeOffset	false	Um mapeamento de um deslocamento de tempo no manifesto para uma lista de anúncios a serem inseridos no momento.

vod CreativeOffset

Esta seção descreve as propriedades dos registros do ADS vodCreativeOffset.

Registros e propriedades do ADS CreativeOffset

Propriedade	Tipo	Obrigatório	Description
adContent	objeto do tipo adContent	true	Informações sobre o conteúdo do anúncio inserido.
creativeUniqueId	string	true	O identificador exclusivo do anúncio, usado como uma chave para transcodificação. Este é o campo de ID do criador na resposta VAST, se disponível. Caso contrário, será o URL de mezanino do anúncio.
trackingEvents	objeto do tipo trackingEvents	true	Os URLs de beacon de rastreamento para os vários eventos de rastreamento do anúncio. As chaves são os nomes de eventos, e os valores são uma lista de URLs de beacon.
transcodedAdDuration	número	true	A duração do anúncio, calculada a partir do ativo transcodificado.

Propriedade	Tipo	Obrigatório	Description
uri	string	true	O URL da versão de mezanino do anúncio, que é a entrada para o transcodificador.
vastDuration	número	true	A duração do anúncio, conforme analisada na resposta VAST.

Esquema JSON de log do ADS

A seguir está uma lista do esquema JSON para o registro do AWS Elemental MediaTailor ADS.

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "http://amazon.com/elemental/midas/mms/adsLogSchema.json",
  "type": "object",
  "title": "AWS Elemental MediaTailor ADS Log JSON Schema",
  "required": [
    "eventType",
    "eventTimestamp",
    "requestId",
    "sessionType",
    "eventDescription",
    "awsAccountId",
    "customerId",
    "originId",
    "sessionId"
  ],
  "additionalProperties": false,
  "properties": {
    "eventType": {
      "$id": "#/properties/eventType",
      "type": "string",
      "description": "The code for the event that triggered this log message. Example: <code>VAST_RESPONSE</code>.",
      "examples": [
        "FILLED_AVAIL"
      ]
    }
  }
}
```

```

    ],
    },
    "eventTimestamp": {
      "$id": "#/properties/eventTimestamp",
      "type": "string",
      "description": "The date and time of the event.",
      "examples": [
        "1970-01-01T00:00:00Z"
      ],
      "format": "date-time"
    },
    "requestId": {
      "$id": "#/properties/requestId",
      "type": "string",
      "description": "The MediaTailor request ID, which you can use to correlate
multiple log entries for the same request.",
      "examples": [
        "c7c7ae8c-a61e-44e0-8efd-7723995337a1"
      ],
      "pattern": "^(.*)$"
    },
    "sessionType": {
      "$id": "#/properties/sessionType",
      "type": "string",
      "enum": [
        "HLS",
        "DASH"
      ],
      "description": "The player's stream type."
    },
    "eventDescription": {
      "$id": "#/properties/eventDescription",
      "type": "string",
      "description": "A short description of the event that triggered this log message,
provided by the MediaTailor service. By default, this is empty. Example: <code>Got
VAST response</code>.",
      "default": "",
      "examples": [
        "Got VAST response"
      ],
      "pattern": "^(.*)$"
    },
    "awsAccountId": {
      "$id": "#/properties/awsAccountId",

```

```

    "type": "string",
    "description": "The AWS account ID for the MediaTailor configuration that was
used for the session."
  },
  "customerId": {
    "$id": "#/properties/customerId",
    "type": "string",
    "description": "The hashed version of the AWS account ID, which you can use to
correlate multiple log entries.",
    "pattern": "^(.*)$"
  },
  "originId": {
    "$id": "#/properties/originId",
    "type": "string",
    "description": "The configuration name from the MediaTailor configuration. This
is different from the video content source, which is also part of the configuration.",
    "examples": [
      "external-canary-dash-serverside-reporting-onebox"
    ],
    "pattern": "^(.*)$"
  },
  "sessionId": {
    "$id": "#/properties/sessionId",
    "type": "string",
    "description": "The unique numeric identifier that MediaTailor assigned to the
player session. All requests that a player makes for a session have the same session
ID. Example: <code>e039fd39-09f0-46b2-aca9-9871cc116cde</code>.",
    "examples": [
      "120b9873-c007-40c8-b3db-0f1bd194970b"
    ],
    "pattern": "^(.*)$"
  },
  "avail": {
    "$id": "#/properties/avail",
    "type": "object",
    "title": "avail",
    "description": "Information about an avail that MediaTailor fills with ads.
Currently, for the <code>FILLED_AVAIL</code> event type, this is the plan created by
MediaTailor when it first encounters the avail. How the avail is eventually filled may
vary from this plan, depending on how the content plays out. ",
    "required": [
      "creativeAds",
      "originAvailDuration",
      "filledDuration",

```

```

    "fillRate",
    "driftMillisecondsAtAvailStart",
    "numAds",
    "slateAd",
    "availId"
  ],
  "additionalProperties": false,
  "properties": {
    "originAvailDuration": {
      "$id": "#/properties/avail/originAvailDuration",
      "type": "number",
      "description": "The duration of the avail as specified in the content stream
from the origin (<code>CUE_OUT</code> or <code>SCTE</code>).",
    },
    "filledDuration": {
      "$id": "#/properties/avail/filledDuration",
      "type": "number",
      "description": "The sum of the durations of all the ads inserted into the
avail."
    },
    "fillRate": {
      "$id": "#/properties/avail/fillRate",
      "type": "number",
      "description": "The rate at which the ads fill the avail duration, from 0.0
(for 0%) to 1.0 (for 100%).",
    },
    "driftMillisecondsAtAvailStart": {
      "$id": "#/properties/avail/driftMillisecondsAtAvailStart",
      "type": "number",
      "description": "The cumulative drift at the beginning of this avail. A
positive value implies that we are moving away from the live edge, a negative value
implies that we are moving towards the live edge."
    },
    "creativeAds": {
      "$id": "#/properties/avail/creativeAds",
      "type": "array",
      "description": "The ads that MediaTailor inserted into the avail.",
      "items": {
        "type": "object",
        "title": "creativeAd",
        "description": "Information about a single inserted ad.",
        "required": [
          "uri",
          "creativeUniqueId",

```

```

        "adSystem",
        "adContent",
        "trackingEvents",
        "vastDuration",
        "transcodedAdDuration"
    ],
    "additionalProperties": false,
    "properties": {
        "uri": { "$ref": "#/definitions/adMezzanineUri" },
        "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
        "adSystem": { "$ref": "#/definitions/adSystem" },
        "adContent": { "$ref": "#/definitions/adContent" },
        "trackingEvents": { "$ref": "#/definitions/trackingEvents" },
        "vastDuration": { "$ref": "#/definitions/vastDuration" },
        "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" }
    }
},
"numAds": {
    "$id": "#/properties/avail/numAds",
    "type": "number",
    "description": "The number of ads that MediaTailor inserted into the avail."
},
"slateAd": {
    "$id": "#/properties/avail/slateAd",
    "type": ["object", "null"],
    "title": "slateAd",
    "description": "Information about the slate ad, which MediaTailor uses to
fill any unfilled segments in the avail.",
    "additionalProperties": false,
    "required": [
        "uri",
        "creativeUniqueId",
        "adContent",
        "transcodedAdDuration"
    ],
    "properties": {
        "uri": { "$ref": "#/definitions/adMezzanineUri" },
        "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
        "adContent": { "$ref": "#/definitions/adContent" },
        "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" }
    }
},
"availId": {

```

```

    "$id": "#/properties/avail/availId",
    "type": "string",
    "description": "The unique identifier for this avail. For HLS, this is the
media sequence number where the avail begins. For DASH, this is the period ID."
  },
  "skippedAds": {
    "$id": "#/properties/avail/skippedAds",
    "type": "array",
    "description": "The ads that MediaTailor didn't insert, for reasons like
<code>TRANSCODE_IN_PROGRESS</code> and <code>TRANSCODE_ERROR</code>.",
    "items": {
      "type": "object",
      "title": "skippedAd",
      "description": "Information about a single skipped ad.",
      "required": [
        "creativeUniqueId",
        "adMezzanineUrl",
        "skippedReason",
        "vastDuration"
      ],
      "additionalProperties": false,
      "properties": {
        "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
        "adMezzanineUrl": {
          "type": "string",
          "description": "The mezzanine URL of the skipped ad."
        },
        "skippedReason": {
          "type": "string",
          "description": "The code that indicates why the ad wasn't inserted.
Example: <code>TRANSCODE_IN_PROGRESS</code>."
        },
        "vastDuration": { "$ref": "#/definitions/vastDuration" },
        "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" },
        "targetVariant": {
          "type": "object",
          "title": "targetVariant",
          "description": "The target variant of the source content. This key is
present when an ad wasn't inserted due to the source content containing a variant that
could not match to any variants present in this ad.",
          "required": [
            "mediaProtocol",
            "mediaType",
            "bitrate",

```



```
        "mediaResolution",
        "codecs"
    ],
    "additionalProperties": false,
    "properties": {
        "mediaProtocol": {
            "type": "string",
            "description": "The media protocol of this variant, such as HLS.",
            "enum": [
                "HLS",
                "DASH"
            ]
        },
        "mediaType": {
            "type": "array",
            "description": "The media type of this variant, such as VIDEO.",
            "items": {
                "type": "string",
                "enum": [
                    "VIDEO",
                    "AUDIO",
                    "SUBTITLES",
                    "TRICK_PLAY"
                ]
            },
            "description": "Media type, such as VIDEO."
        }
    },
    "bitrate": {
        "$ref": "#/definitions/bitrate"
    },
    "mediaResolution": {
        "type": "object",
        "title": "mediaResolution",
        "description": "The media resolution of this variant.",
        "required": [
            "width",
            "height"
        ],
        "additionalProperties": false,
        "properties": {
            "width": {
                "$ref": "#/definitions/width"
            },
            "height": {
```

```

        "$ref": "#/definitions/height"
      }
    }
  },
  "codecs": {
    "type": "array",
    "description": "The codecs of this variant.",
    "items": {
      "type": "string",
      "description": "Codec, such as avc1."
    }
  }
}
},
"adBreakTrackingEvents": {
  "$id": "#properties/avail/adBreakTrackingEvents",
  "type": "object",
  "title": "adBreakTrackingEvents",
  "description": "VMAP/ad break tracking events and corresponding URL",
  "properties": {
    "items": {
      "$ref": "#/definitions/adBreakTrackingEvents"
    }
  }
}
},
"vastResponse": {
  "$id": "#/properties/vastResponse",
  "type": "object",
  "title": "vastResponse",
  "description": "Information about the VAST response that MediaTailor received
from the ADS.",
  "required": [
    "version",
    "vastAds",
    "errors",
    "nonLinearAdsList"
  ],
  "additionalProperties": false,

```

```

    "properties": {
      "version": {
        "$id": "#/properties/vastResponse/version",
        "type": "string",
        "description": "The VAST specification version, parsed from the
<code>version</code> attribute of the <code>VAST</code> tag in the response.",
        "examples": [
          "3.0"
        ],
        "pattern": "^(.*)$"
      },
      "vastAds": {
        "$id": "#/properties/vastResponse/vastAds",
        "type": "array",
        "description": "The ads parsed from the VAST response.",
        "items": {
          "$ref": "#/definitions/vastAd"
        }
      },
      "errors": {
        "$id": "#/properties/vastResponse/errors",
        "type": "array",
        "description": "The error URLs parsed from the <code>Error</code> tags in the
VAST response.",
        "items": {
          "type": "string",
          "description": "A single error URL."
        }
      },
      "nonLinearAdsList": {
        "$id": "#/properties/vastResponse/nonLinearAds",
        "type": "array",
        "description": "A list of NonLinearAds as they are read from the VAST
response.",
        "items": {
          "$ref": "#/definitions/nonLinearAds"
        }
      }
    },
    "vastAd": {
      "$ref": "#/definitions/vastAd"
    }
  },

```

```

"vodVastResponseTimeOffset": {
  "$id": "#/properties/vodVastResponseTimeOffset",
  "type": "number",
  "description": "The VMAP specific time offset for VOD ad insertion.",
  "examples": [
    5.0
  ]
},

"vodCreativeOffsets": {
  "$id": "#/properties/vodCreativeOffsets",
  "type": "object",
  "title": "vodCreativeOffsets",
  "description": "A map that indicates the time offsets in the manifest where
MediaTailor will insert avails, based on the VMAP response.",
  "additionalProperties": {
    "type": "array",
    "$id": "#/properties/vodCreativeOffsets/entry",
    "description": "A mapping from a time offset in the manifest to a list of ads
to insert at this time.",
    "items": {
      "type": "object",
      "$id": "#/properties/vodCreativeOffsets/entry/items",
      "title": "vodCreativeOffset",
      "description": "The list of ads to insert at the specified time offset.",
      "additionalProperties": false,
      "required": [
        "uri",
        "creativeUniqueId",
        "vastDuration",
        "transcodedAdDuration",
        "adContent",
        "trackingEvents"
      ],
      "properties": {
        "uri": { "$ref": "#/definitions/adMezzanineUri" },
        "creativeUniqueId": { "$ref": "#/definitions/creativeUniqueId" },
        "vastDuration": { "$ref": "#/definitions/vastDuration" },
        "transcodedAdDuration": { "$ref": "#/definitions/transcodedAdDuration" },
        "adContent": { "$ref": "#/definitions/adContent" },
        "trackingEvents": { "$ref": "#/definitions/trackingEvents" }
      }
    }
  ]
}

```

```

    }
  },

  "adsRequestUrl": {
    "$id": "#/properties/adsRequestUrl",
    "type": "string",
    "description": "The full URL of the ADS request made by MediaTailor."
  },

  "adMarkers": {
    "$id": "#/properties/adMarkers",
    "type": "string",
    "description": "Found Ad Marker in the Manifest."
  },

  "segmentationUpid": {
    "$id": "#/properties/segmentationUpid",
    "type": "string",
    "description": "Value of segmentation upid parsed from ad markers in manifest."
  },

  "segmentationTypeId": {
    "$id": "#/properties/segmentationTypeId",
    "type": "integer",
    "description": "Value of segmentation typeId parsed from ad markers in manifest."
  },

  "requestHeaders": {
    "$id": "#/properties/requestHeaders",
    "type": "array",
    "description": "The headers that MediaTailor included with the ADS request.
Typically, the logs include these when a request to the ADS fails, to help with
troubleshooting.",
    "items": {
      "type": "object",
      "title": "requestheaders",
      "description": "The name and value for a single header included in the ADS
request.",
      "required": [
        "name",
        "value"
      ],
      "additionalProperties": false,
      "properties": {
        "name": {
          "type": "string",
          "description": "The name of the header."
        }
      }
    }
  },

```

```

        "value": {
            "type": "string",
            "description": "The value of the header."
        }
    }
},

"originalTargetUrl": {
    "$id": "#/properties/originalTargetUrl",
    "type": "string",
    "description": "The old URL to which MediaTailor was going to make a request."
},
"prefetchScheduleName": {
    "$id": "#/properties/prefetchScheduleName",
    "type": "string",
    "description": "The name of the prefetch schedule associated with this ad
event.",
    "examples": [
        "PrefetchScheduleName"
    ],
    "pattern": "^(.*)$"
},
"updatedTargetUrl": {
    "$id": "#/properties/updatedTargetUrl",
    "type": "string",
    "description": "The new URL to which MediaTailor is making a request."
},

"rawAdsResponse": {
    "$id": "#/properties/rawAdsResponse",
    "type": "string",
    "description": "Paginated ADS response as it's exactly returned to MediaTailor."
},
"rawAdsResponseIndex": {
    "$id": "#/properties/rawAdsResponseIndex",
    "type": "integer",
    "description": "Integer value denoting this rawAdsResponse's index into the
full ADS response. This value is used to order the paginated messages for this ADS
response."
},

"eventId": {
    "$id": "#/properties/eventId",

```

```

    "type": "string",
    "description": "Unique identifier per hook invocation. Use this field to
correlate hook-level and function-level events for the same execution.",
    "examples": [
        "5dc6f040-0f72-4e8c-a64e-25eeef62708c"
    ]
},
"functionId": {
    "$id": "#/properties/functionId",
    "type": "string",
    "description": "The identifier of the function. On hook events, this is the
function configured on the hook. On function events, this is the individual function
within the executor.",
    "examples": [
        "my_first_function"
    ]
},
"functionType": {
    "$id": "#/properties/functionType",
    "type": "string",
    "enum": [
        "CUSTOM_OUTPUT",
        "HTTP_REQUEST",
        "SEQUENTIAL_EXECUTOR",
        "CONCURRENT_EXECUTOR"
    ],
    "description": "The type of function."
},
"executionTimeMs": {
    "$id": "#/properties/executionTimeMs",
    "type": "integer",
    "description": "Execution time in milliseconds."
},
"status": {
    "$id": "#/properties/status",
    "type": "string",
    "enum": [
        "SUCCESS",
        "ERROR"
    ],
    "description": "Hook execution outcome. Present on HOOK_SUMMARY events only."
},
"errorType": {
    "$id": "#/properties/errorType",

```

```
    "type": "string",
    "enum": [
      "INTERNAL_ERROR",
      "SYNTAX_ERROR",
      "RESOURCE_LIMIT_ERROR",
      "RESTRICTION_ERROR",
      "TIMEOUT_ERROR",
      "VALIDATION_ERROR"
    ],
    "description": "Structured error classification. Present on error events only."
  },
  "cause": {
    "$id": "#/properties/cause",
    "type": "string",
    "description": "Error detail string describing the specific failure."
  },
  "input": {
    "$id": "#/properties/input",
    "type": "object",
    "description": "Before-values of variables targeted by output expressions."
  },
  "output": {
    "$id": "#/properties/output",
    "type": "object",
    "description": "Values produced by the function. Present on FUNCTION_COMPLETED events only."
  },
  "http": {
    "$id": "#/properties/http",
    "type": "object",
    "description": "HTTP request and response details for HTTP_REQUEST functions.",
    "properties": {
      "request": {
        "type": "object",
        "description": "The HTTP request details.",
        "properties": {
          "url": { "type": "string", "description": "The request URL." },
          "method": { "type": "string", "description": "The HTTP method." },
          "headers": {
            "type": "array",
            "description": "The request headers.",
            "items": {
              "type": "object",
              "properties": {
```



```

        "name": { "type": "string" },
        "value": { "type": "string" }
      }
    },
    "body": { "type": ["string", "null"], "description": "The request body." }
  },
  "responseTimeMs": { "type": ["integer", "null"], "description": "HTTP response
time in milliseconds." },
  "response": {
    "type": ["object", "null"],
    "description": "The HTTP response details. Null when the server was never
reached.",
    "properties": {
      "statusCode": { "type": ["integer", "null"], "description": "The HTTP
status code. Null when the server was never reached." },
      "headers": {
        "type": ["array", "null"],
        "description": "The response headers.",
        "items": {
          "type": "object",
          "properties": {
            "name": { "type": "string" },
            "value": { "type": "string" }
          }
        }
      }
    }
  },
  "body": { "type": ["string", "null"], "description": "The response body." }
},
"truncated": { "type": "boolean", "description": "True when request or response
body was truncated to fit within log event size limits." }
}
},

```

"__COMMENT_oneOf": "The oneOf section defines subtypes for our events. Subtypes can have different rules, including which fields are required. For more information, see <https://json-schema.org/understanding-json-schema/reference/combining.html#oneof> ",

```

"oneOf": [
  { "$ref": "#/definitions/eventAdMarkersFound" },
  { "$ref": "#/definitions/eventNonAdMarkerFound" },

```

```

{ "$ref": "#/definitions/eventMakingAdsRequest" },
{ "$ref": "#/definitions/eventModifiedTargetUrl" },
{ "$ref": "#/definitions/eventRawAdsResponse" },
{ "$ref": "#/definitions/eventVastResponse" },
{ "$ref": "#/definitions/eventFilledAvail" },
{ "$ref": "#/definitions/eventFilledOverlayAvail" },
{ "$ref": "#/definitions/eventErrorFiringBeaconFailed" },
{ "$ref": "#/definitions/eventWarningNoAdvertisements" },
{ "$ref": "#/definitions/eventUnknownHost" },
{ "$ref": "#/definitions/eventErrorAdsTimeout" },
{ "$ref": "#/definitions/eventErrorVastMissingOverlays" },
{ "$ref": "#/definitions/eventPlannedAvail" },
{ "$ref": "#/definitions/eventEmptyVastResponse" },
{ "$ref": "#/definitions/eventEmptyVmapResponse" },
{ "$ref": "#/definitions/eventErrorUnknown" },
{ "$ref": "#/definitions/eventVastRedirect" },
{ "$ref": "#/definitions/eventRedirectedVastResponse" },
{ "$ref": "#/definitions/eventErrorAdsMissingImpression" },
{ "$ref": "#/definitions/eventErrorAdsResponseParse" },
{ "$ref": "#/definitions/eventErrorAdsInvalidResponse" },
{ "$ref": "#/definitions/eventErrorDisallowedHost" },
{ "$ref": "#/definitions/eventPersonalizationDisabled" },
{ "$ref": "#/definitions/eventWarningDynamicVariableSubFailed" },
{ "$ref": "#/definitions/eventVodTimeBasedAvailPlanVastResponseForOffset" },
{ "$ref": "#/definitions/eventVodTimeBasedAvailPlanSuccess" },
{ "$ref": "#/definitions/eventPreAdsRequestHookSummary" },
{ "$ref": "#/definitions/eventPreAdsRequestHookError" },
{ "$ref": "#/definitions/eventPreAdsRequestFunctionCompleted" },
{ "$ref": "#/definitions/eventPreAdsRequestFunctionError" }
],

"definitions": {
  "eventAdMarkersFound": {
    "$id": "#/definitions/eventAdMarkersFound",
    "required": [
      "eventType",
      "adMarkers"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "AD_MARKER_FOUND"
      }
    }
  }
}

```

```
},
"eventNonAdMarkerFound": {
  "$id": "#/definitions/eventNonAdMarkerFound",
  "required": [
    "eventType",
    "adMarkers"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "NON_AD_MARKER_FOUND"
    }
  }
},
"eventMakingAdsRequest": {
  "$id": "#/definitions/eventMakingAdsRequest",
  "required": [
    "eventType",
    "adsRequestUrl"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "MAKING_ADS_REQUEST"
    }
  }
},
"eventModifiedTargetUrl": {
  "$id": "#/definitions/eventModifiedTargetUrl",
  "required": [
    "eventType",
    "originalTargetUrl",
    "updatedTargetUrl"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "MODIFIED_TARGET_URL"
    }
  }
},
"eventRawAdsResponse": {
```

```

    "$id": "#/definitions/eventRawAdsResponse",
    "required": [
        "eventType",
        "rawAdsResponse",
        "rawAdsResponseIndex"
    ],
    "properties": {
        "eventType": {
            "type": "string",
            "const": "RAW_ADS_RESPONSE"
        }
    }
},

```

"eventVastResponse": {
 "\$id": "#/definitions/eventVastResponse",
 "_comment": "NOTE: the vastResponse property should ideally be marked as a required field for this event, but unfortunately, in the case of an empty vast response, we currently emit an EMPTY_VAST_RESPONSE followed by a VAST_RESPONSE, and the vastResponse property is not present in the latter. We need to fix this so that we don't emit both of those events in the empty response case, and update this schema to flag vastResponse as required for VAST_RESPONSE.",

```

    "required": [
        "eventType"
    ],
    "properties": {
        "eventType": {
            "type": "string",
            "const": "VAST_RESPONSE"
        }
    }
},

```

```

"eventFilledAvail": {
    "$id": "#/definitions/eventFilledAvail",
    "required": [
        "eventType",
        "avail"
    ],
    "properties": {
        "eventType": {
            "type": "string",
            "const": "FILLED_AVAIL"
        }
    }
}

```

```
    }
  },

  "eventFilledOverlayAvail": {
    "$id": "#/definitions/eventFilledOverlayAvail",
    "required": [
      "eventType",
      "avail"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "FILLED_OVERLAY_AVAIL"
      }
    }
  },

  "eventErrorVastMissingOverlays": {
    "$id": "#/definitions/eventErrorVastMissingOverlays",
    "required": [
      "eventType",
      "adsRequestUrl",
      "requestHeaders"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_VAST_MISSING_OVERLAYS"
      }
    }
  },

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

```
    }
  },

  "eventWarningNoAdvertisements": {
    "$id": "#/definitions/eventWarningNoAdvertisements",
    "required": [
      "eventType"
    ],
    "_comment": "We should really have a more descriptive error field for these events",
    "properties": {
      "eventType": {
        "type": "string",
        "const": "WARNING_NO_ADVERTISEMENTS"
      }
    }
  },

  "eventUnknownHost": {
    "$id": "#/definitions/eventUnknownHost",
    "required": [
      "eventType",
      "requestHeaders"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_UNKNOWN_HOST"
      }
    }
  },

  "eventErrorAdsTimeout": {
    "$id": "#/definitions/eventErrorAdsTimeout",
    "required": [
      "eventType",
      "adsRequestUrl",
      "requestHeaders"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_ADS_TIMEOUT"
      }
    }
  }
}
```

```
    }
  },

  "eventPlannedAvail": {
    "$id": "#/definitions/eventPlannedAvail",
    "required": [
      "eventType"
    ],
    "_comment": "TODO: Flesh this out as we implement it",
    "properties": {
      "eventType": {
        "type": "string",
        "const": "PLANNED_AVAIL"
      }
    }
  },

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

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

  "eventErrorUnknown": {
```

```

    "$id": "#/definitions/eventErrorUnknown",
    "required": [
        "eventType"
    ],
    "_comment": "TODO: we should have a field for the exception message or
something",
    "properties": {
        "eventType": {
            "type": "string",
            "const": "ERROR_UNKNOWN"
        }
    }
},

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

"eventRedirectedVastResponse": {
    "$id": "#/definitions/eventRedirectedVastResponse",
    "required": [
        "eventType"
    ],
    "properties": {
        "eventType": {
            "type": "string",
            "const": "REDIRECTED_VAST_RESPONSE"
        }
    },
    "_comment": "NOTE that the property vastResponse is not required because empty
vast responses do not contain a vastResponse."
},

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

```



```
    "required": [
      "eventType"
    ],
    "_comment": "We should have a field with an error message here",
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_ADS_RESPONSE_PARSE"
      }
    }
  },

  "eventErrorAdsInvalidResponse": {
    "$id": "#/definitions/eventErrorAdsInvalidResponse",
    "required": [
      "eventType",
      "additionalInfo"
    ],
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_ADS_INVALID_RESPONSE"
      }
    }
  },

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

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

```
    "properties": {
      "eventType": {
        "type": "string",
        "const": "ERROR_DISALLOWED_HOST"
      }
    }
  },

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

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

  "eventVodTimeBasedAvailPlanVastResponseForOffset": {
    "$id": "#/definitions/eventVodTimeBasedAvailPlanVastResponseForOffset",
    "required": [
      "eventType",
      "vastResponse"
    ],
    "properties": {
      "eventType": {
        "type": "string",
```

```

        "const": "VOD_TIME_BASED_AVAIL_PLAN_VAST_RESPONSE_FOR_OFFSET"
    }
}
},

"eventVodTimeBasedAvailPlanSuccess": {
    "$id": "#/definitions/eventVodTimeBasedAvailPlanSuccess",
    "required": [
        "eventType",
        "vodCreativeOffsets"
    ],
    "properties": {
        "eventType": {
            "type": "string",
            "const": "VOD_TIME_BASED_AVAIL_PLAN_SUCCESS"
        }
    }
},

"creativeUniqueId": {
    "type": "string",
    "description": "The unique identifier for the ad, used as a key for transcoding.
This is the ID field for the creative in the VAST response, if available. Otherwise,
it's the mezzanine URL of the ad. "
},

"adSystem": {
    "type": "string",
    "description": "The value of the <code>AdSystem</code> tag in the VAST response.
"
},

"vastDuration": {
    "type": "number",
    "description": "The duration of the ad, as parsed from the VAST response."
},

"transcodedAdDuration": {
    "type": "number",
    "description": "The duration of the ad, calculated from the transcoded asset."
},

"adContent": {
    "$id": "#/properties/adContent",

```

```

    "type": ["object", "null"],
    "title": "adContent",
    "description": "Information about the content of the inserted ad.",
    "additionalProperties": false,
    "properties": {
      "adPlaylistUris": {
        "$id": "#/properties/adContent/adPlaylistUris",
        "type": "object",
        "title": "adPlaylistUris",
        "description": "The mapping from the origin manifest for a variant to the ad
manifest for the variant. For DASH, this contains a single entry, because all variants
are represented in a single DASH manifest. ",
        "additionalProperties": {
          "$id": "#/properties/adContent/adPlaylistUris/adPlaylistUri",
          "type": "string",
          "description": "The URL of the ad manifest for the specific variant."
        }
      }
    }
  },
  "adMezzanineUri": {
    "type": "string",
    "description": "The URL of the mezzanine version of the ad, which is the input to
the transcoder."
  },
  "bitrate": {
    "type": "integer",
    "examples": [
      533
    ],
    "description": "The bitrate."
  },
  "width": {
    "type": "integer",
    "examples": [
      1280
    ],
    "description": "Width in pixels."
  },
  "height": {
    "type": "integer",
    "examples": [

```

```

    720
  ],
  "description": "Height in pixels."
},

"trackingEvents": {
  "type": "object",
  "title": "trackingEvents",
  "description": "The tracking beacon URLs for the various tracking events for the
ad. The keys are the event names, and the values are a list of beacon URLs.",

  "additionalProperties": {
    "type": "array",
    "description": "The list of beacon URLs for the specified tracking event
(impression, complete, and so on)",
    "items": {
      "type": "string",
      "description": "The beacon URLs for this tracking event."
    }
  }
},

"nonLinearAds": {
  "$id": "#/properties/nonLinearAds",
  "type": "object",
  "title": "nonLinearAds",
  "description": "A NonLinearAds as it appears in the VAST response.",
  "required": [
    "nonLinearAdList",
    "nonLinearTrackingEvents"
  ],
  "properties": {
    "nonLinearAdList": {
      "type": "array",
      "description": "List of non linear ads as they exist within one
NonLinearAds.",
      "items": {
        "type": "object",
        "title": "nonLinearAdList",
        "description": "List of NonLinearAd as they are parsed from its parent
NonLinearAds.",
        "properties": {
          "nonLinearAdId": {
            "type": "string",

```

```
    "description": "Ad ID of this non linear ad."
  },
  "nonLinearAdSystem": {
    "type": "string",
    "description": "Ad system of this non linear ad's parent Inline ad."
  },
  "nonLinearAdTitle": {
    "type": "string",
    "description": "Ad title of this non linear ad's parent Inline ad."
  },
  "nonLinearCreativeId": {
    "type": "string",
    "description": "Creative ID of this non linear ad's parent Creative
ad."
  },
  "nonLinearCreativeAdId": {
    "type": "string",
    "description": "Creative ad ID of this non linear ad."
  },
  "nonLinearCreativeSequence": {
    "type": "string",
    "description": "Creative sequence of this non linear ad."
  },
  "nonLinearWidth": {
    "type": "string",
    "description": "Width of this non linear ad."
  },
  "nonLinearHeight": {
    "type": "string",
    "description": "Height of this non linear ad."
  },
  "nonLinearExpandedWidth": {
    "type": "string",
    "description": "Expanded width of this non linear ad."
  },
  "nonLinearExpandedHeight": {
    "type": "string",
    "description": "Expanded height of this non linear ad."
  },
  "nonLinearScalable": {
    "type": "boolean",
    "description": "Boolean denoting if this non linear ad is scalable."
  },
  "nonLinearMaintainAspectRatio": {
```

```
    "type": "boolean",
    "description": "Boolean denoting if aspect ratio should be maintained
for this non linear ad."
  },
  "nonLinearMinSuggestedDuration": {
    "type": "string",
    "description": "Min suggested duration for this non linear ad."
  },
  "nonLinearApiFramework": {
    "type": "string",
    "description": "API framework for this non linear ad's parent Inline
ad."
  },
  "nonLinearStaticResource": {
    "type": "string",
    "description": "Static resource for this non linear ad."
  },
  "nonLinearStaticResourceCreativeType": {
    "type": "string",
    "description": "Static Resource creative type for this non linear ad."
  },
  "nonLinearIFrameResource": {
    "type": "string",
    "description": "I-Frame resource for this non linear ad."
  },
  "nonLinearHtmlResource": {
    "type": "string",
    "description": "HTML resource for this non linear ad."
  },
  "nonLinearAdParameters": {
    "type": "string",
    "description": "Ad parameters for this non linear ad."
  },
  "nonLinearClickThrough": {
    "type": "string",
    "description": "Click Through data for this non linear ad."
  },
  "nonLinearClickTracking": {
    "type": "string",
    "description": "Click Tracking data for this non linear ad."
  },
  "nonLinearClickTrackingId": {
    "type": "string",
    "description": "Click Tracking ID for this non linear ad."
  }
```

```

    }
  }
}
},
"nonLinearTrackingEvents": { "$ref": "#/definitions/trackingEvents" },
"extensions": {
  "$id": "#/properties/nonLinearAds/extensions",
  "type": "array",
  "description": "Extensions that exist for this NonLinearAds.",
  "items": {
    "$id": "#/properties/nonLinearAds/extensions/items",
    "type": "object",
    "title": "Extensions",
    "description": "Extensions found in non linear ads",
    "additionalProperties": false,
    "properties": {
      "extensionType": {
        "$id": "#/properties/nonLinearAds/extensions/extensionType",
        "type": "string",
        "description": "The value of the extension type attribute of the
<code>Extensions</code> tag.",
        "examples": [
          "FreeWheel"
        ]
      },
      "extensionContent": {
        "$id": "#/properties/nonLinearAds/extensions/extensionContent",
        "type": "string",
        "description": "The extension content attribute of the
<code>Extensions</code> tag.",
        "examples": [
          "progressive"
        ]
      }
    }
  }
}
},
"adBreakTrackingEvents": {
  "$id": "#/properties/adBreakTrackingEvents",
  "type": "object",
  "title": "adBreakTrackingEvents",
  "description": "These are all VMAP ad break tracking events.",

```



```

    "additionalProperties": {
      "type": "array",
      "description": "VMAP/ad break tracking events and corresponding URL",
      "items": {
        "type": "string",
        "description": "The beacon URLs for this tracking event."
      }
    }
  },
  "vastAd": {
    "$id": "#/properties/vastAd",
    "type": "object",
    "title": "vastAd",
    "description": "Information about a single ad parsed from the VAST response.",
    "required": [
      "vastAdId",
      "adSystem",
      "adTitle",
      "creativeId",
      "creativeAdId",
      "duration",
      "vastMediaFiles",
      "trackingEvents"
    ],
    "additionalProperties": false,
    "properties": {
      "vastAdId": {
        "$id": "#/properties/vastAd/vastAdId",
        "type": "string",
        "description": "The value of the id attribute of the <code>Ad</code> tag in
the VAST response",
        "examples": [
          "ad1"
        ]
      },
      "adSystem": {"$ref": "#/definitions/adSystem" } ,
      "adTitle": {
        "$id": "#/properties/vastAd/adTitle",
        "type": "string",
        "description": "The media files that are available for the ad in the VAST
response.",
        "examples": [
          "External NCA1C1L1 LinearInlineSkippable"
        ]
      }
    }
  }
}

```

```
    },
    "creativeId": {
      "$id": "#/properties/vastAd/creativeId",
      "type": "string",
      "description": "The value of the id attribute of the <code>Creative</code> tag in the VAST response.",
      "examples": [
        "creative1"
      ]
    },
    "creativeAdId": {
      "$id": "#/properties/vastAd/creativeAdId",
      "type": "string",
      "description": "The value of the adId attribute of the <code>Creative</code> tag in the VAST response."
    },
    "duration": {
      "$id": "#/properties/vastAd/duration",
      "type": "number",
      "description": "The approximate duration of the ad, based on the <code>duration</code> tag in the <code>linear</code> element of the VAST response.",
      "examples": [
        30,
        30.0
      ]
    },
    "vastMediaFiles": {
      "$id": "#/properties/vastAd/vastMediaFiles",
      "type": "array",
      "description": "The list of available media files for the ad in the VAST response.",
      "items": {
        "$id": "#/properties/vastAd/vastMediaFiles/items",
        "type": "object",
        "title": "vastMediaFile",
        "description": "Information about a media file for the ad.",
        "required": [
          "uri",
          "id",
          "delivery",
          "type",
          "apiFramework",
          "width",
          "height",
```

```

        "bitrate"
    ],
    "additionalProperties": false,
    "properties": {
        "uri": { "$ref": "#/definitions/adMezzanineUri" },
        "id": {
            "$id": "#/properties/vastAd/vastMediaFiles/items/properties/id",
            "type": "string",
            "description": "The value of the id attribute of the <code>MediaFile</code> tag.",
            "examples": [
                "GDFP"
            ]
        },
        "delivery": {
            "$id": "#/properties/vastAd/vastMediaFiles/items/properties/delivery",
            "type": "string",
            "description": "The protocol used for the media file, set to either progressive or streaming.",
            "examples": [
                "progressive"
            ]
        },
        "type": {
            "$id": "#/properties/vastAd/vastMediaFiles/items/properties/type",
            "type": "string",
            "description": "The MIME type of the media file, taken from the type attribute of the <code>MediaFile</code> tag.",
            "examples": [
                "video/mp4"
            ]
        },
        "apiFramework": {
            "$id": "#/properties/vastAd/vastMediaFiles/items/properties/apiFramework",
            "type": "string",
            "description": "The API framework needed to manage the media file. Example: <code>VPAID</code>."
        },
        "width": {
            "$ref": "#/definitions/width"
        },
        "height": {
            "$ref": "#/definitions/height"
        }
    }
}

```

```

        },
        "bitrate": {
            "$ref": "#/definitions/bitrate"
        }
    }
},
"trackingEvents": { "$ref": "#/definitions/trackingEvents" },
"vastAdTagUri": {
    "$id": "#/properties/vastAd/vastAdTagUri",
    "type": "string",
    "description": "The VMAP-specific redirect URI for an ad.",
    "examples": [
        "https://ads.redirect.com/redirect1"
    ]
}
},
},
"eventPreAdsRequestHookSummary": {
    "$id": "#/definitions/eventPreAdsRequestHookSummary",
    "required": [
        "eventType",
        "eventId",
        "functionId",
        "executionTimeMs",
        "status"
    ],
    "properties": {
        "eventType": {
            "type": "string",
            "const": "PRE_ADS_REQUEST_HOOK_SUMMARY"
        }
    }
},
},
"eventPreAdsRequestHookError": {
    "$id": "#/definitions/eventPreAdsRequestHookError",
    "required": [
        "eventType",
        "eventId",
        "functionId",
        "executionTimeMs",
        "errorType",

```

```
    "cause"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "PRE_ADS_REQUEST_HOOK_ERROR"
    }
  }
},

"eventPreAdsRequestFunctionCompleted": {
  "$id": "#/definitions/eventPreAdsRequestFunctionCompleted",
  "required": [
    "eventType",
    "eventId",
    "functionId",
    "functionType",
    "executionTimeMs"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "PRE_ADS_REQUEST_FUNCTION_COMPLETED"
    }
  }
},

"eventPreAdsRequestFunctionError": {
  "$id": "#/definitions/eventPreAdsRequestFunctionError",
  "required": [
    "eventType",
    "eventId",
    "functionId",
    "functionType",
    "executionTimeMs",
    "errorType",
    "cause"
  ],
  "properties": {
    "eventType": {
      "type": "string",
      "const": "PRE_ADS_REQUEST_FUNCTION_ERROR"
    }
  }
}
```

```
}  
}  
}
```

AWS Elemental MediaTailor descrição dos registros de manifesto e tipos de eventos

As seções a seguir descrevem os registros que são MediaTailor emitidos para descrever eventos com o servidor de origem ao solicitar e receber um manifesto. São ManifestService registros.

Tópicos

- [ManifestService eventos](#)
- [Propriedades dos registros do manifesto](#)

ManifestService eventos

Os seguintes eventos são emitidos durante MediaTailor as interações com a origem.

Log	Description
CONFIG_SECURITY_ERROR	A MediaTailor configuração tem um problema de segurança.
CONFIG_SYNTAX_ERROR	A origem e o caminho do ativo resultam em uma URL malformada.
CONNECTION_ERROR	A MediaTailor conexão com a origem foi recusada ou falhou.
GENERATED_MANIFEST	MediaTailor gerou um manifesto. Você deve ter o modo de depuração ativado para receber esses registros. Para obter informações sobre o modo de registro de depuração, incluindo como ativá-lo, consulte Gerando registros de depuração .

Log	Description
HOST_DISALLOWED	MediaTailor não permite solicitações HTTP para esse host.
INCOMPATIBLE_HLS_VERSION	O manifesto usa uma versão HLS incompatível. MediaTailor requer a versão 3 ou superior.
INVALID_SINGLE_PERIOD_DASH_MANIFEST	O manifesto DASH de período único é inválido. MediaTailor está passando pelo manifesto DASH de período único.
IO_ERROR	MediaTailor encontrou um erro de E/S durante a comunicação com a origem.
LAST_PERIOD_MISSING_AUDIO	O último ponto no manifesto do DASH não contém todo o áudio AdaptationSets devido ao desalinhamento do áudio ou vídeo de origem. Para evitar problemas de reprodução, adie a publicação do último período até pelo menos a próxima solicitação.
LAST_PERIOD_MISSING_AUDIO_WARNING	O último ponto no manifesto do DASH não contém todo o áudio AdaptationSets devido ao desalinhamento do áudio ou vídeo de origem. Optar por publicar (sem atrasar) o último período. A falta de áudio pode causar problemas de reprodução.
MANIFEST_ERROR	A solicitação do MediaTailor manifesto falhou.
NO_MASTER_OR_MEDIA_PLAYLIST	A resposta de origem não contém uma playlist primária ou playlist de mídia.
NO_MASTER_PLAYLIST	A resposta de origem não contém a playlist primária esperada.
NO_MEDIA_PLAYLIST	A resposta de origem não contém a playlist de mídia esperada.

Log	Description
ORIGIN_MANIFEST	MediaTailor buscou um manifesto de origem. Você deve ter o modo de depuração ativado para receber esses registros. Para obter informações sobre o modo de registro de depuração, incluindo como ativá-lo, consulte Gerando registros de depuração .
PARSING_ERROR	A origem não consegue analisar a solicitação do manifesto.
PRE_SESSION_INIT_FUNCTION_COMPLETED	Uma função individual no gancho de inicialização da pré-sessão foi concluída. Esse é um tipo de evento opcional.
PRE_SESSION_INIT_FUNCTION_ERROR	Uma função individual no gancho de inicialização da pré-sessão falhou.
PRE_SESSION_INIT_HOOK_ERROR	A execução do gancho de inicialização da pré-sessão falhou.
PRE_SESSION_INIT_HOOK_SUMMARY	Resumo da execução do gancho de inicialização antes da sessão, incluindo status de sucesso ou erro. Esse é um tipo de evento opcional.
SCTE35_PARSING_ERROR	MediaTailor não consegue analisar o Signal Binary elemento no manifesto.
SESSION_INITIALIZED	Uma sessão foi inicializada. Você deve ter o modo de depuração ativado para receber esses registros. Para obter informações sobre o modo de registro de depuração, incluindo como ativá-lo, consulte Gerando registros de depuração .

Log	Description
TIMEOUT_ERROR	O tempo limite da solicitação do MediaTailor manifesto expirou.
TRACKING_RESPONSE	MediaTailor serviu uma resposta de rastreamento. Você deve ter o modo de depuração ativado para receber esses registros. Para obter informações sobre o modo de registro de depuração, incluindo como ativá-lo, consulte Gerando registros de depuração .
UNKNOWN_ERROR	MediaTailor encontrou um erro desconhecido.
UNKNOWN_HOST	O anfitrião é desconhecido.
UNSUPPORTED_SINGLE_PERIOD_DASH_MANIFEST	O manifesto DASH de período único não é suportado. MediaTailor está passando pelo manifesto DASH de período único.

Propriedades dos registros do manifesto

Esta seção descreve as propriedades dos registros do manifesto.

Propriedade	Tipo	Obrigatório
awsAccountId	string	true
eventTimestamp	string	true
originId	string	true
customerId	string	false
eventType	string	false
sessionId	string	false

Propriedade	Tipo	Obrigatório
originRequestUrl	string	false
mediaTailorPath	string	false
requestId	string	false
responseBody	string	false
sessionType	string (valores legais: [DASH, HLS])	false
requestNextToken	string	false
eventDescription	string	false
errorType	string	false
eventId	string	false
executionTimeMs	integer	false
functionId	string	false
functionType	string	false
http	objeto	false
input	objeto	false
output	objeto	false
status	string	false
assetPath	string	false
originFullUrl	string	false
originPrefixUrl	string	false
additionalInfo	string	false

Propriedade	Tipo	Obrigatório
cause	string	false
response	string	false
httpCode	string	false
errorMessage	string	false
adAdsResponse	string	false
adAdsRawResponse	string	false
adAdsRequest	string	false
adNumNewAvails	string	false
generatedMediaPlay list	string	false
requestStartTime	string	false
requestEndTime	string	false
requestStartTimeEp ochMillis	string	false
requestEndTimeEpoc hMillis	string	false

AWS Elemental MediaTailor transcodificar registros, descrição e tipos de eventos

As seções a seguir descrevem os registros que são MediaTailor emitidos para descrever eventos com o serviço de transcodificação ao preparar criativos para a criação de anúncios. São TranscodeService registros.

Tópicos

- [TranscodeService eventos](#)
- [Propriedades de registros de transcodificação](#)

TranscodeService eventos

Os eventos a seguir são emitidos durante MediaTailor as interações durante a transcodificação dos anúncios.

Log	Description
IMPORT_ERROR	MediaTailor encontrou um erro interno durante um trabalho de importação (para anúncios pré-condicionados). Usando um conjunto vazio de anúncios.
INITIALIZED	MediaTailor inicializou uma tarefa de transcodificação ou uma tarefa de importação (para anúncios pré-condicionados).
INTERNAL_ERROR	MediaTailor encontrou um erro interno. Usando um conjunto vazio de anúncios.
MISSING_VARIANTS	MediaTailor não foi possível transcodificar o anúncio devido à falta de variantes. Usando um conjunto vazio de anúncios.
PROFILE_NOT_FOUND	MediaTailor não foi possível transcodificar o anúncio devido à falta de um perfil para transcodificar. Usando um conjunto vazio de anúncios.
TRANSCODE_COMPLETED	A transcodificação de vídeo está completa. O anúncio pode ser usado para inserção de anúncios.
TRANSCODE_ERROR	MediaTailor encontrou um erro interno durante um trabalho de transcodificação. Usando um conjunto vazio de anúncios.

Log	Description
TRANSCODE_IN_PROGRESS	A transcodificação do vídeo está em andamento. O vídeo transcodificado não está pronto. Usando um conjunto vazio de anúncios.

Propriedades de registros de transcodificação

Esta seção descreve as propriedades dos registros de transcodificação.

Propriedade	Tipo	Obrigatório	Descrição
awsAccountId	string	true	O ID da AWS conta para a MediaTailor configuração que foi usada para a sessão.
eventTimestamp	string	true	A data e a hora do evento.
originId	string	true	O nome da MediaTailor configuração. Isso é diferente da fonte de conteúdo de vídeo, que também faz parte da configuração.
eventType	string	false	O código do evento que acionou essa mensagem de log. Exemplo: TRANSCODE_ERROR .
eventDescription	string	false	Uma breve descrição do evento que acionou essa mensagem de

Propriedade	Tipo	Obrigatório	Descrição
			log, fornecida pelo MediaTailor serviço. Por padrão, permanece em branco.
sessionId	string	false	O identificador numérico exclusivo MediaTailor atribuído à sessão do jogador. Todas as solicitações que um player faz para uma sessão têm o mesmo ID de sessão. Exemplo: e039fd39-09f0-46b2-aca9-9871cc116cde .
creativeUniqueId	string	false	O identificador exclusivo do anúncio criativo que está sendo transcodificado.
profileName	string	false	
adUri	string	false	O URI do anúncio criativo.
transcodeRequestId	string	false	O identificador exclusivo dessa solicitação de transcodificação.

Propriedade	Tipo	Obrigatório	Descrição
cacheStatus	string	false	Indica se o MediaTailor ou anúncio transcodificado foi armazenado em cache.

Usando registros vendidos para enviar AWS Elemental MediaTailor logs

Você pode usar registros vendidos para maior flexibilidade e controle sobre onde entregar os registros MediaTailor emitidos pela sua configuração de reprodução.

Com os registros vendidos, MediaTailor envia todas as atividades de registro associadas a uma configuração para o Amazon CloudWatch Logs. CloudWatch Em seguida, o Logs envia a porcentagem de registros que você especifica para o destino escolhido. Os destinos compatíveis são um grupo de CloudWatch logs do Amazon Logs, um bucket do Amazon S3 ou um stream do Amazon Data Firehose.

Como os registros vendidos estão disponíveis com preços de desconto por volume, você pode ver uma economia de custos em comparação ao envio de registros diretamente para o CloudWatch Logs. Para preços, consulte Vended Logs na guia Logs em [Amazon CloudWatch Pricing](#).

Para usar registros vendidos, você deve fazer o seguinte:

1. [Adicionar permissões](#).
2. [Crie destinos de entrega de registros](#).
3. [Configurar a entrega de CloudWatch registros em Logs](#).
4. [Ativar logins de venda MediaTailor](#).

Para obter mais informações sobre registros vendidos, consulte [Habilitar o registro de AWS serviços](#) no guia do usuário de CloudWatch registros. MediaTailor suporta V2 de registros vendidos.

Etapa 1: adicionar permissões para entrega de MediaTailor registros

A pessoa que está configurando os registros vendidos deve ter permissões para criar o destino da entrega, configurar a entrega dos registros e habilitar os logins vendidos. MediaTailor Use as

políticas a seguir para garantir que você tenha as permissões apropriadas para configurar registros vendidos.

Políticas para CloudWatch registros e destinos de entrega

As seções a seguir no Guia do usuário do Amazon CloudWatch Logs fornecem as políticas que permitem que você trabalhe com CloudWatch registros no Logs e em seus destinos de entrega. Se você enviar registros para vários locais, poderá combinar as declarações de política em uma política em vez de criar várias políticas.

- [Registros enviados para CloudWatch Logs](#)
- [Logs enviados ao Amazon S3](#)
- [Registros enviados para o Firehose](#)

Política para configuração a partir do console

Se você estiver configurando a entrega de registros vendidos por meio do console em vez da API ou AWS CLI, deverá ter as seguintes permissões adicionais em sua política.

Política para logins vendidos MediaTailor

Para criar, visualizar ou modificar a entrega de registros vendidos MediaTailor, você deve ter as seguintes permissões em sua política.

Para obter informações sobre como adicionar permissões e trabalhar com políticas, consulte [Identity and Access Management para AWS Elemental MediaTailor](#).

Etapa 2: criar destinos de entrega para MediaTailor registros

Crie os recursos para os quais seus registros serão enviados. Registre o ARN do recurso para uso na configuração da entrega do log em uma etapa posterior.

CloudWatch Registra o destino de entrega do grupo de registros

Use uma das opções a seguir para ajudar a criar um grupo de registros.

- Para o console, consulte [Criar um grupo de CloudWatch registros em Logs](#) no Guia do usuário do Amazon CloudWatch Logs.
- Para a API, consulte [CreateLogGroup](#) Referência da API Amazon CloudWatch Logs.
- Para SDKs e CLI, [consulte CreateLogGroup Usar com AWS um SDK](#) ou AWS CLI no Guia do usuário do CloudWatch Amazon Logs.

Destino de entrega do bucket Amazon S3

Use uma das opções a seguir para ajudar a criar um bucket.

- Para o console, SDKs e CLI, [consulte Criar um](#) bucket no Guia do usuário do Amazon Simple Storage Service.
- Para a API, consulte a Referência [CreateBucket](#) de API do Amazon Simple Storage Service.

Destino de entrega do stream Firehose

Para obter ajuda na criação de um stream, consulte [Create a Firehose stream from console](#) no Amazon Data Firehose Developer Guide.

Etapa 3: Habilitar registros vendidos para a configuração de MediaTailor reprodução

Crie ou atualize a configuração de reprodução que enviará os registros para o destino de entrega que você criou na etapa anterior. Registre o nome da configuração para uso na configuração da entrega do log em uma etapa posterior.

- Para habilitar registros vendidos por meio do console, usando [Criar uma configuração](#) ou [Editando as configurações](#) editando uma configuração para acessar as configurações de registro. Para estratégias de registro, escolha Vended logs.
- Para habilitar registros vendidos por meio da API, você deve ter uma configuração existente. Use `ConfigureLogsForPlaybackConfiguration` para adicionar a estratégia de registro `Vended logs`.

Se você estiver usando a estratégia de MediaTailor registro antiga de enviar registros diretamente para o CloudWatch Logs e quiser migrar para registros vendidos, consulte. [Migrando a estratégia de registro](#)

Important

Se você alterar a estratégia de registro de registros antigos CloudWatch para registros vendidos, MediaTailor fará essa alteração assim que você salvar as atualizações. Você deixará de receber registros até ter configurado totalmente o registro vendido.

Etapa 4: configurar a entrega de CloudWatch registros em Registros

Em CloudWatch Registros, você precisa criar três elementos para representar as partes da entrega do registro. Esses elementos são descritos detalhadamente [CreateDelivery](#) na Referência da API Amazon CloudWatch Logs. As etapas de alto nível para configurar a entrega com a API CloudWatch Logs são as seguintes.

Para configurar a entrega de CloudWatch registros em Logs (API)

1. Use [PutDeliverySource](#) para adicionar a fonte dos registros.

A `DeliverySource` representa a configuração de reprodução que está gerando os registros. Você precisa do nome da configuração de reprodução para criar o `DeliverySource`.

2. Use [PutDeliveryDestination](#) para adicionar o destino onde os registros serão gravados.

A `DeliveryDestination` representa o destino da entrega. Você precisa do ARN do grupo de registros, bucket ou stream para criar o `DeliveryDestination`.

3. Use [PutDeliveryDestinationPolicy](#) se você estiver entregando registros em várias contas.

Se o destino da entrega estiver em uma conta diferente da configuração de reprodução, você precisará de um `DeliveryDestinationPolicy`. Essa política permite que o CloudWatch Logs entregue registros para `DeliveryDestination`.

4. Use [CreateDelivery](#) para vincular o `DeliverySource` ao `DeliveryDestination`.

A `Delivery` representa a conexão entre `DeliverySource` e `DeliveryDestination`.

Migrando seu AWS Elemental MediaTailor estratégia de registro

Se você alterar a estratégia de registro de registros antigos CloudWatch para registros vendidos, MediaTailor fará essa alteração assim que você salvar as atualizações. Para evitar interrupções no fluxo de trabalho de registro, use as etapas a seguir para migrar sua estratégia de registro.

1. Siga as etapas conforme descritas em [Usando registros vendidos](#). Para [Ativar logins de venda MediaTailor](#), selecione as duas estratégias de registro (Vended logs e Legacy CloudWatch).

MediaTailor enviará registros por meio de registros vendidos e diretamente para o CloudWatch Logs.

2. Faça as alterações necessárias em seu fluxo de trabalho que dependem da sua estratégia de registro e do destino de entrega.
3. Revisite [Ativar logins de venda MediaTailor](#) e remova o Legacy CloudWatch das estratégias de registro.

Writing AWS Elemental MediaTailor registra diretamente no Amazon CloudWatch Logs

MediaTailor produz registros que contêm informações detalhadas sobre a atividade da sessão e as interações do servidor de decisão de anúncios e os grava na Amazon CloudWatch. Os registros fornecem uma descrição sequencial da atividade que ocorre durante a sessão.

MediaTailor também pode usar registros vendidos para flexibilidade na entrega de registros e preços com desconto por volume. Para obter informações sobre registros vendidos, consulte [Usando registros vendidos](#).

Tópicos

- [Permissões para Amazon CloudWatch Logs](#)
- [Registro “As Run” para AWS Elemental MediaTailor Montagem do canal](#)
- [AWS Elemental MediaTailor Análise de registros do ADS no Amazon CloudWatch Logs Insights](#)

Permissões para Amazon CloudWatch Logs

Use AWS Identity and Access Management (IAM) para criar uma função que dê AWS Elemental MediaTailor acesso à Amazon CloudWatch. Você deve executar essas etapas para que CloudWatch os registros sejam publicados em sua conta. CloudWatch publica automaticamente métricas para sua conta.

Para permitir o MediaTailor acesso a CloudWatch

1. Abra o console do IAM em <https://console.aws.amazon.com/iam/>.
2. No painel de navegação do console do IAM, escolha Perfis e, em seguida, Criar perfil.
3. Escolha o tipo de função Outro tipo de AWS conta.
4. Em ID da conta, insira o ID AWS da sua conta.

5. Selecione Require external ID (Exigir ID externo) e digite **Midas**. Essa opção adiciona automaticamente uma condição à política de confiança que só permite ao serviço assumir a função caso a solicitação inclua o `sts:ExternalId` correto.
6. Escolha Próximo: Permissões.
7. Adicione uma política de permissões que especifique quais ações essa função pode concluir. Selecione uma das seguintes opções e escolha Next: Review (Avançar: revisar):
 - `CloudWatchLogsFullAccess` para fornecer acesso total ao Amazon CloudWatch Logs
 - `CloudWatchFullAccess` para fornecer acesso total à Amazon CloudWatch
8. Em Role name (Nome da função), digite **MediaTailorLogger** e escolha Create role (Criar função).
9. Na página Roles (Funções), escolha a função recém-criada.
10. Para atualizar a entidade principal, edite a relação de confiança:
 1. Na página Summary (Resumo) da função, escolha a guia Trust relationship (Relação de confiança).
 2. Selecione Editar relação de confiança.
 3. No documento de política, altere o principal para o MediaTailor serviço. A aparência deve ser semelhante a esta:

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

Toda a política deve ser lida da seguinte maneira:

4. Selecione Atualizar política de confiança.

Registro “As Run” para AWS Elemental MediaTailor Montagem do canal

O registro As Run, no grupo de CloudWatch MediaTailor/Channel/AsRunLog registros, mostra informações sobre programas e intervalos publicitários enquanto eles são reproduzidos.

Quando você cria um canal, o registro As Run é desativado por padrão. Usando o console ou o AWS Command Line Interface (AWS CLI), você pode ativar e desativar o estado de registro As Run para cada canal da sua conta.

Quando você ativa o registro As Run, cria MediaTailor automaticamente uma função vinculada ao serviço que permite MediaTailor gravar e gerenciar o registro As Run na sua conta do CloudWatch Logs. Para obter mais informações sobre funções vinculadas ao serviço, consulte [Usando funções vinculadas a serviços para MediaTailor](#).

 Note

Atualmente, o As Run Log suporta apenas o programa padrão. Por enquanto, ele não suporta o AlternateMedia criado pelas regras do programa. Isso significa que atualmente ele não gera o As Run Log para AlternateMedia.

Tópicos

- [Ativando o registro As Run](#)
- [Desativando o registro As Run](#)

Ativando o registro As Run

Para ativar o registro As Run, especifique o nome do canal e ative o tipo de log As Run para esse canal.

Console

Para habilitar o log As Run ao criar um canal

1. Faça login no Console de gerenciamento da AWS e abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. Na barra de navegação, escolha Criar canal.
4. Nos painéis Definir detalhes do canal, Configurar saídas e Controle de acesso, configure seu canal conforme desejado.
5. No painel Controle de acesso, escolha Avançar.
6. No painel Registro, em Tipos de registro, selecione Ativar como execução para ativar o registro Como execução.

Para ativar o registro As Run ao atualizar um canal

Note

Se o canal estiver em execução no momento, você deve primeiro interromper esse canal antes de poder atualizá-lo. Depois de parar o canal, você pode escolher Ações > Editar para começar a atualizar o canal.

1. Faça login no Console de gerenciamento da AWS e abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. Escolha o canal que você deseja atualizar para ativar o registro As Run para.
4. Selecione Actions (Ações) > Edit (Editar).
5. Nos painéis Definir detalhes do canal, Configurar saídas e Controle de acesso, atualize a configuração do canal conforme desejado.
6. No painel Controle de acesso, escolha Avançar.
7. No painel Registro, em Tipos de registro, selecione Ativar como execução para ativar o registro Como execução.

Para habilitar o log As Run a partir do Registro em log tab

Note

Se o canal estiver em execução no momento, você deverá usar a guia Registro em vez de escolher Ações > Editar para ativar o registro As Run.

1. Faça login no Console de gerenciamento da AWS e abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. Escolha o canal para o qual você deseja habilitar o registro As Run.
4. Na barra de navegação abaixo do nome do canal, escolha Registro.
5. Em Registro > Tipos de registro, selecione Como execução para ativar o registro Como execução.

AWS Command Line Interface (AWS CLI)

Para habilitar o registro As Run

Execute o comando [configure-logs-for-channel](#) e especifique os valores apropriados para os parâmetros necessários.

Este exemplo está formatado para Linux, macOS ou Unix e usa o caractere de continuação de linha “barra invertida (\)” para melhorar a legibilidade.

```
$ aws mediatailor configure-logs-for-channel \  
--channel-name MyChannel \  
--log-types AS_RUN
```

Este exemplo foi formatado para Microsoft Windows e usa o caractere de continuação de linha circunflexo (^) para melhorar a legibilidade.

```
C:\> aws mediatailor configure-logs-for-channel ^  
--channel-name MyChannel ^  
--log-types AS_RUN
```

Em que:

- *MyChannel* é o nome do canal que você possui e para o qual deseja ativar o registro As Run.

Se o comando for executado com sucesso, você receberá um resultado semelhante ao seguinte.

```
{  
  "ChannelName": "MyChannel",  
  "LogTypes": [  
    "AS_RUN"  
  ]  
}
```

Desativando o registro As Run

Para desativar o registro As Run para um canal que o tenha ativado, especifique o nome do canal e desative o tipo de log As Run para esse canal.

Console

Para desativar o registro As Run ao atualizar um canal

Note

Se o canal estiver em execução no momento, você deve primeiro interromper esse canal antes de poder atualizá-lo. Depois de parar o canal, você pode escolher Ações > Editar para começar a atualizar o canal.

1. Faça login no Console de gerenciamento da AWS e abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. Escolha o canal que você deseja atualizar para ativar o registro As Run para.
4. Selecione Actions (Ações) > Edit (Editar).
5. Nos painéis Definir detalhes do canal, Configurar saídas e Controle de acesso, atualize a configuração do canal conforme desejado.
6. No painel Controle de acesso, escolha Avançar.
7. No painel Registro, em Tipos de registro, desmarque Habilitar como execução para desativar o registro Como Execução.

Para desativar o registro As Run do Registro em log tab

Note

Se o canal estiver em execução no momento, você deverá usar a guia Registro em vez de escolher Ações > Editar para desativar o registro As Run.

1. Faça login no Console de gerenciamento da AWS e abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel de navegação, escolha Montagem de canais > Canais.
3. Escolha o canal para o qual você deseja desativar o registro As Run.
4. Na barra de navegação abaixo do nome do canal, escolha Registro.

5. Em Registro > Tipos de registro, desmarque As run para desativar o log As Run.

AWS Command Line Interface (AWS CLI)

Para desativar o registro As Run

Execute o comando [configure-logs-for-channel](#) e especifique os valores apropriados para os parâmetros necessários.

Este exemplo está formatado para Linux, macOS ou Unix e usa o caractere de continuação de linha “barra invertida (\)” para melhorar a legibilidade.

```
$ aws mediatailor configure-logs-for-channel \  
--channel-name MyChannel \  
--log-types
```

Este exemplo foi formatado para Microsoft Windows e usa o caractere de continuação de linha circunflexo (^) para melhorar a legibilidade.

```
C:\> aws mediatailor configure-logs-for-channel ^  
--channel-name MyChannel ^  
--log-types
```

Em que:

- *MyChannel* é o nome do canal que você possui e do qual deseja desativar o registro As Run.

Se o comando for executado com sucesso, você receberá um resultado semelhante ao seguinte.

```
{  
  "ChannelName": "MyChannel",  
  "LogTypes": []  
}
```

AWS Elemental MediaTailor Análise de registros do ADS no Amazon CloudWatch Logs Insights

Você pode visualizar e consultar AWS Elemental MediaTailor os registros do servidor de decisão de anúncios (ADS) usando o Amazon CloudWatch Logs Insights. MediaTailor envia registros de eventos

CloudWatch para condições normais de processamento e erro. Os logs seguem um esquema JSON. Por meio do CloudWatch Logs Insights, você pode selecionar registros por período de tempo e, em seguida, executar consultas neles.

Para obter informações gerais, consulte [Analisar dados de registro com o CloudWatch Logs Insights](#).

 Note

Para acessar os registros, você precisa de permissões para acessar a Amazon CloudWatch. Para instruções, consulte [Permissões para Amazon CloudWatch Logs](#).

Para visualizar e consultar registros do ADS usando o CloudWatch console

1. Abra o CloudWatch console em <https://console.aws.amazon.com/cloudwatch/>.
2. No painel de navegação, em Logs, escolha Insights.
3. Na barra de pesquisa, digite **eAdDec**, na lista suspensa, selecione. MediaTailor/AdDecisionServerInteractions
4. (Opcional) Ajuste o período que você deseja estudar.
5. (Opcional) Altere a consulta na caixa de diálogo. Para orientação geral, consulte [Sintaxe de consulta do CloudWatch Logs Insights](#). Para obter exemplos de consultas no ADS do MediaTailor, consulte [Consultando os registros do ADS](#).
6. Selecione Executar consulta. A consulta pode levar alguns segundos, durante os quais Cancel (Cancelar) é exibido no lugar de Run query (Executar consulta).
7. (Opcional) Para exportar os resultados como um arquivo CSV, escolha Actions (Ações) e, depois, selecione Download query results (CSV) (Fazer download de resultados da consulta [CSV]).

 Note

O console limita o número de registros que ele retorna nos resultados da consulta e exporta. Portanto, para dados em massa, use a API, o AWS Command Line Interface (AWS CLI) ou um SDK.

Tópicos

- [Consultando os registros do ADS](#)

Consultando os registros do ADS

CloudWatch O Logs Insights fornece um rico conjunto de opções para consultar seus registros. Para obter informações detalhadas sobre a sintaxe de consulta, consulte Sintaxe de [consulta do CloudWatch Logs Insights](#). Esta seção fornece exemplos de consultas comuns para você começar a executar suas consultas de logs do ADS. Todas as consultas são executadas nos logs da configuração de intervalo de tempo atual.

A consulta a seguir recupera todas as informações dos logs do ADS.

```
fields @timestamp, eventType, sessionId, requestId, @message
| sort sessionId, @timestamp asc
```

A consulta a seguir recupera todas as solicitações para o ADS. Essa consulta mostra uma forma de recuperar o conteúdo do cabeçalho da solicitação para MediaTailor os registros.

```
fields @timestamp, adsRequestUrl, requestHeaders.0.value as @userAgent,
requestHeaders.1.value as @xForwardedFor, sessionId, requestId
| filter eventType = "MAKING_ADS_REQUEST"
| sort @timestamp asc
```

A consulta a seguir recupera os anúncios MediaTailor inseridos em uma determinada sessão.

```
fields @timestamp, sessionId, requestId, @message
| filter eventType = "FILLED_AVAIL"
| sort @timestamp asc
```

A consulta a seguir recupera os URLs de rastreamento que MediaTailor ligaram em nome do jogador.

```
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
```

A consulta a seguir recupera informações de uma sessão de reprodução específica, filtrando os resultados por sessionId.

```
fields @timestamp, eventType, sessionId, requestId, @message
| filter sessionId = "0aaf6507-c6f9-4884-bfe7-f2f841cb8195"
| sort @timestamp asc
```

A consulta a seguir recupera informações de uma única solicitação, filtrando os resultados por `requestId`.

```
fields @timestamp, eventType, sessionId, requestId, @message
| filter requestId = "f5d3cf39-6258-4cf1-b3f6-a34ff8bf641d"
| sort @timestamp asc
```

A consulta a seguir recupera uma contagem de entradas de log para cada tipo de evento que foi registrado.

```
fields eventType
| stats count() as @eventCount by eventType
```

A consulta a seguir recupera o ID de disponibilidade e a lista de anúncios ignorados para todas as disponibilidades que ignoraram anúncios.

```
fields avail.availId
| parse @message '"skippedAds":[*]' as @skippedAdsList
| filter ispresent(@skippedAdsList)
```

Controlando o volume de AWS Elemental MediaTailor logs

MediaTailor Às vezes, os registros da sessão de inserção de anúncios são detalhados. Para reduzir os custos de registro, você pode definir a porcentagem de registros de sessão que são MediaTailor enviados para o Amazon CloudWatch Logs. Por exemplo, se sua configuração de reprodução tiver 1.000 sessões de inserção de anúncios e você definir um valor percentual ativado de 60, MediaTailor enviará registros de 600 das sessões para CloudWatch o Logs. MediaTailor decide aleatoriamente para qual das sessões enviar registros. Se quiser ver os registros de uma sessão específica, você pode usar o [modo de registro de depuração](#).

Quando você define uma porcentagem de registro, cria MediaTailor automaticamente um papel vinculado ao serviço que concede MediaTailor as permissões necessárias para gravar CloudWatch registros na sua conta. Para obter informações sobre como MediaTailor usa funções vinculadas a serviços, consulte [Usando funções vinculadas a serviços para MediaTailor](#)

Criando uma configuração de log

Para controlar a porcentagem de registros de sessão que MediaTailor gravam em CloudWatch registros, você cria uma configuração de registro para sua configuração de reprodução. Ao criar uma configuração de log, você especifica um nome de configuração de reprodução e um valor percentual ativado.

Console

Para criar uma configuração de log para um existente configuração de reprodução

1. Faça login no Console de gerenciamento da AWS e abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel Configuração de reprodução, selecione a configuração de reprodução para a qual você gostaria de definir a configuração de registro.
3. Escolha Editar.
4. Em Configuração do log, especifique um valor percentual ativado.

Para criar uma configuração de log para um nova configuração de reprodução

- Siga o procedimento em [Configuração de log](#).

AWS Command Line Interface (AWS CLI)

Para criar uma configuração de log para uma configuração de reprodução existente

Para criar uma configuração de log usando o AWS CLI, execute o comando [configure-logs-for-playback-configuration](#) e especifique os valores apropriados para os parâmetros necessários.

Este exemplo está formatado para Linux, macOS ou Unix e usa o caractere de continuação de linha “barra invertida (\)” para melhorar a legibilidade.

```
$ aws mediatailor configure-logs-for-playback-configuration \
--percent-enabled 10 \
--playback-configuration-name MyPlaybackConfiguration
```

Este exemplo foi formatado para Microsoft Windows e usa o caractere de continuação de linha circunflexo (^) para melhorar a legibilidade.

```
C:\> aws mediatailor configure-logs-for-playback-configuration ^  
--percent-enabled 10 ^  
--playback-configuration-name MyPlaybackConfiguration
```

Em que:

- *percent-enabled* é a porcentagem dos registros da sessão de configuração de reprodução que são MediaTailor enviados para o CloudWatch Logs.
- *playback-configuration-name* é o nome da configuração de reprodução para a qual definir as configurações de log.

Se o comando for executado com sucesso, você receberá um resultado semelhante ao seguinte.

```
{  
  "PercentEnabled": 10,  
  "PlaybackConfigurationName": "MyPlaybackConfiguration"  
}
```

Para criar uma configuração de log para uma nova configuração de reprodução

- Use a `configure-logs-for-playback-configuration` opção para o comando [put-playback-configuration](#).

Desativando uma configuração de log

Depois de criar uma configuração de log, você não pode excluí-la — você só pode desativá-la. Para desativar a configuração do log, defina o valor da porcentagem ativada como 0 com o MediaTailor console ou a API. Isso desativa todo o registro de sessão dessa configuração de reprodução.

Se você quiser excluir a função vinculada ao serviço que MediaTailor usa para as configurações de registro em sua conta, primeiro desative todas as suas configurações de registro. Para obter informações sobre como excluir a função vinculada ao serviço, consulte [Usando funções vinculadas a serviços para MediaTailor](#)

Console

Para desativar a configuração de log em uma configuração de reprodução

1. Faça login no Console de gerenciamento da AWS e abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. No painel Configuração de reprodução, selecione a configuração de reprodução na qual você gostaria de desativar a configuração de registro.
3. Escolha Editar.
4. Em Configuração do log, defina o valor da porcentagem ativada como 0. Isso desativa todo o registro de sessão para essa configuração de reprodução.
5. Selecione Salvar.

AWS Command Line Interface (AWS CLI)

Para desativar uma configuração de log

- Defina o `percent-enabled` valor para 0 usar o comando [configure-logs-for-playback-configuration](#).

Filtrar AWS Elemental MediaTailor registros e eventos

Os registros emitidos de uma configuração de reprodução em MediaTailor incluem informações sobre uma variedade de atividades que acontecem durante a sessão de reprodução. Essas atividades são identificadas no tipo de evento dos registros. Muitos eventos são registrados por padrão. Para ajudar a controlar o custo dos registros na Amazon CloudWatch, você pode especificar os registros que são MediaTailor emitidos.

MediaTailor fornece controle sobre a filtragem de registros para que você possa fazer o seguinte:

- Especifique os eventos de registro que você deseja excluir dos registros
- Ative o registro de respostas brutas do servidor de decisão de anúncios (ADS)

Você pode definir essas preferências de filtragem de registros de forma independente para cada sessão de reprodução ou como padrão para todas as sessões de reprodução de uma configuração de reprodução.

- Para filtrar registros por sessão, inclua parâmetros de consulta na solicitação de inicialização da sessão de reprodução.
- Para filtrar registros com base na configuração por reprodução, use o MediaTailor console ou a API para indicar suas preferências nas configurações de reprodução.

As seções a seguir fornecem instruções para ativar a filtragem de registros em sessões e configurações de reprodução.

Per-session filtros de registro

Para definir um nível personalizado de detalhes do registro para cada sessão, anexe os seguintes parâmetros à sua solicitação inicial de sessão de reprodução do lado do servidor ou do lado do cliente. Adicione valores aos parâmetros para representar os eventos que você deseja incluir ou excluir, em um formato delimitado por vírgula:

- `aws.adsInteractionLogPublishOptInEventTypes` para receber registros de interações específicas do servidor de decisão de anúncios (ADS).
- `aws.adsInteractionLogExcludeEventTypes` para parar de receber registros de interações específicas do ADS.
- `aws.manifestServiceLogExcludeEventTypes` para parar de receber registros de interações específicas do serviço de manifesto.

Para obter uma lista dos tipos de registros e eventos que MediaTailor emitem [Registros de manifestos](#), consulte [Registros do ADS](#), e [Registros de transcodificação](#)

Se você não passar por nenhum parâmetro de consulta para filtragem de registros, MediaTailor grava todos os registros no seu destino de entrega.

Exemplo inicialização da sessão do lado do servidor com filtros de log

Para excluir GENERATED_MANIFEST PARSING_ERROR eventos de seus registros de manifesto e MAKING_ADS_REQUEST dos registros do ADS, a solicitação de inicialização da sessão ficaria assim:

```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/index.m3u8?  
aws.logMode=DEBUG&aws.manifestServiceLogExcludeEventTypes=GENERATED_MANIFEST,PARSING_ERROR&aws.
```

Para ativar os registros brutos do seu ADS, inclua o RAW_ADS_RESPONSE valor do AdsInteractionPublishOptInEventType parâmetro:


```
GET <mediatailorURL>/v1/master/<hashed-account-id>/<origin-id>/index.m3u8?
aws.adsInteractionPublishOptInEventType=RAW_ADS_RESPONSE
```

Exemplo inicialização da sessão do lado do cliente com filtros de log

Para excluir eventos de log durante a inicialização da sessão do lado do cliente, inclua `availSuppression` e digite parâmetros de registro na solicitação POST do seu cliente para. MediaTailor Para obter mais informações sobre como criar uma solicitação de sessão de reprodução no lado do cliente, consulte [Client-side rastreamento de anúncios](#). O exemplo a seguir exclui `CONFIG_SECURITY_ERROR` `PARSING_ERROR` eventos de seus registros de manifesto e `MAKING_ADS_REQUEST` dos registros do ADS.

```
POST parent.m3u8
{
  "adsInteractionLog": {
    ...
    "excludeEventTypes": [
      "MAKING_ADS_REQUEST"
    ]
  },
  "manifestServiceLog": {
    ...
    "excludeEventTypes": [
      "GENERATED_MANIFEST",
      "PARSING_ERROR"
    ]
  },
  "logMode": "DEBUG"
}
```

Para ativar os registros brutos do seu ADS, inclua o `RAW_ADS_RESPONSE` valor do `publishOptInEventTypes` parâmetro:

```
POST parent.m3u8
{
  "adsInteractionLog": {
    "publishOptInEventTypes": ["RAW_ADS_RESPONSE"],
    "excludeEventTypes": [
      "MAKING_ADS_REQUEST"
    ]
  },
}
```

```
"manifestServiceLog": {  
    ...  
    "excludeEventTypes": [  
        "GENERATED_MANIFEST",  
        "PARSING_ERROR"  
    ],  
},  
"logMode": "DEBUG"  
}
```

Per-playback filtros de registro de configuração

Use as configurações de reprodução para definir os tipos de eventos de log que são MediaTailor emitidos como padrão a partir dessa configuração de reprodução. MediaTailor usa essas configurações padrão de filtragem de registros para todas as sessões que não incluem parâmetros de consulta de filtragem na solicitação inicial da sessão.

Você pode optar por fazer o seguinte:

- Receba registros de interações específicas do servidor de decisão de anúncios (ADS).
- Exclua registros de interações específicas do ADS.
- Exclua registros de interações específicas do serviço manifesto.

Para definir essas configurações no MediaTailor console, consulte [Criar uma configuração](#). Para a MediaTailor API, consulte [PutPlaybackConfiguration](#) na Referência AWS Elemental MediaTailor da API.

Para obter uma lista dos tipos de registros e eventos que MediaTailor emitem [Registros de manifestos](#), consulte [Registros do ADS](#), e [Registros de transcodificação](#)

Generating (Gerando) AWS Elemental MediaTailor registros de depuração

Use registros de depuração para solucionar problemas na sessão de reprodução de inserção de MediaTailor anúncios. Para gerar registros de depuração, defina o modo de registro para depurar na solicitação do player. MediaTailor Para relatórios do lado do servidor, defina o modo de registro na solicitação de reprodução. Para relatórios do lado do cliente, defina o modo de log na solicitação de inicialização da sessão.

Quando o modo de log está configurado para depuração, MediaTailor grava todos os tipos de eventos de log em CloudWatch Logs. Os registros fornecem informações sobre os eventos a seguir.

Para obter uma lista completa dos dados produzidos nos registros de depuração, consulte Campos do [registro de depuração](#).

- Interação de origem — Detalhes sobre MediaTailor interações com o servidor de origem. Por exemplo, a resposta do manifesto de origem, o tipo de manifesto e o URL de origem.
- Manifesto gerado — Detalhes sobre a resposta da sessão de reprodução de MediaTailor. Por exemplo, o manifesto que MediaTailor gera.
- Sessão inicializada — Detalhes da inicialização da sessão, como o ID da sessão.

Para personalizar os tipos de eventos de log que você recebe por sessão, consulte [Filtrando registros e eventos](#).

Pré-requisitos

Para definir o modo de registro para depuração, primeiro você precisa conceder MediaTailor permissão para enviar registros para CloudWatch, caso ainda não tenha feito isso. Depois de conceder permissão MediaTailor para acessar CloudWatch, você estará pronto para ativar o modo de registro de depuração. Para obter informações sobre como conceder MediaTailor permissão de acesso, CloudWatch consulte [Configurando permissões para a Amazon CloudWatch](#).

Como definir o modo de log para depuração

Esta seção explica como definir o modo de log para depuração para relatórios do lado do servidor e do lado do cliente.

Server-side relatando

Para relatórios do lado do servidor, inclua o parâmetro e o valor da `?aws.LogMode=DEBUG` consulta na solicitação de GET HTTP reprodução do player para o endpoint HLS ou DASH. MediaTailor [Para obter informações gerais sobre relatórios do lado do servidor, consulte Relatórios. Server-side](#)

Important

O valor DEBUG diferencia letras maiúsculas de minúsculas.

Uma solicitação de reprodução que inclui `?aws.LogMode=DEBUG` tem a seguinte aparência:

Example Solicitação de reprodução para um endpoint HLS

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

Depois de definir o modo de registro para depuração, recomendamos que você verifique se a sessão de registro de depuração está ativa. Para verificar se a sessão de depuração está ativa, verifique se há algum CloudWatch registro para o ID da sessão. O ID da sessão está incluído no endpoint de reprodução que MediaTailor fornece. Para obter mais informações, consulte [Verify that the debug log mode is active for your playback session](#).

Client-side relatando

Para relatórios do lado do cliente, inclua a `logMode` chave e o `DEBUG` valor no corpo da solicitação de inicialização da POST HTTP sessão do seu cliente no MediaTailor endpoint `/v1/session` [Para obter informações gerais sobre relatórios do lado do cliente, consulte Client-Side Relatórios](#).

Important

O valor `DEBUG` diferencia letras maiúsculas de minúsculas.

Depois de definir o modo de log para depuração, recomendamos que você verifique se a sessão de depuração está ativa. Para verificar se a sessão de depuração está ativa, confirme se há um `SESSION_INITIALIZED` evento associado ao ID da sessão nos CloudWatch registros. O ID da sessão está incluído no endpoint de reprodução que MediaTailor fornece. Para obter mais informações, consulte [Verify that the debug log mode is active for your playback session](#).

Máximo de sessões de depuração ativas

Você pode ter no máximo 10 sessões ativas de log de depuração. Quando seu player envia a solicitação de inicialização ou reprodução da sessão para MediaTailor, MediaTailor verifica se o limite foi atingido. Se tiver, MediaTailor verifica se há alguma sessão obsoleta. Uma sessão fica obsoleta se não for acessada em um determinado período de tempo. Para transmissões ao vivo, esse período é de 10 minutos, para transmissões de VOD é de 30 minutos.

Se o limite máximo de sessões ativas do registro de depuração for atingido, os registros de depuração não serão CloudWatch gravados nos registros da sua sessão. Se você não vê registros de depuração nos CloudWatch registros da sua sessão, você pode ter atingido esse limite. Para

confirmar se o limite foi atingido, consulte [Verify that the debug log mode is active for your playback session](#).

Campos de registro de depuração

A tabela a seguir lista os campos do log de depuração nos quais MediaTailor grava. CloudWatch

Campo	Description
<code>awsAccountId</code>	Sua Conta da AWS identidade.
<code>customerId</code>	Seu ID de MediaTailor cliente.
<code>eventTimestamp</code>	O carimbo de data/hora ISO 8601 associado ao evento do registro de depuração.
<code>eventType</code>	O tipo de evento do registro de depuração. Valores: • <code>ORIGIN_INTERACTION</code> — Detalhes sobre MediaTailor interações com o servidor de origem. Por exemplo, a resposta do manifesto de origem, o tipo de manifesto e o URL de origem. • <code>GENERATED_MANIFEST</code> — Detalhes sobre a resposta da sessão de reprodução de MediaTailor. Por exemplo, o manifesto que MediaTailor gera. • <code>SESSION_INITIALIZED</code> — Detalhes da inicialização da sessão, como o ID da sessão.
<code>originRequestUrl</code>	A URL do seu servidor de origem que é recuperada para essa solicitação.
<code>mediaTailorPath</code>	O MediaTailor endpoint que foi chamado, incluindo todos os parâmetros passados MediaTailor na solicitação inicial do manifesto.
<code>requestId</code>	O ID de uma solicitação HTTP específica para MediaTailor.
<code>responseBody</code>	O manifesto no corpo de resposta de MediaTailor. Esse é o manifesto de origem bruta ou o manifesto gerado por MediaTailor.

Campo	Description
sessionId	O ID da sessão de reprodução.
sessionType	O tipo de sessão de reprodução. Valores: HLS, DASH

Leia os registros de depuração

MediaTailor grava os registros de depuração no Amazon CloudWatch Logs. Aplicam-se taxas típicas de CloudWatch registros. Use o CloudWatch Insights para ler os registros de depuração. Para obter informações sobre como usar o CloudWatch Logs Insights, consulte [Análise de dados de log com o CloudWatch Logs Insights](#) no Guia do usuário do AWS CloudWatch Logs.

Note

Os registros de depuração podem levar alguns minutos para aparecer. CloudWatch Se você não visualizar os logs, aguarde alguns minutos e tente novamente. Se você ainda não vê os registros, pode ser que tenha atingido o número máximo de sessões ativas do registro de depuração. Para verificar se esse é o caso, execute uma CloudWatch consulta para ver se houve uma sessão de depuração inicializada para sua sessão de reprodução. Para obter mais informações, consulte [Verify that the debug log mode is active for your playback session](#).

Exemplos

Esta seção inclui exemplos de consultas que você pode usar para ler dados de log de MediaTailor depuração.

Example 1: Verifique se o modo de registro de depuração está ativo para sua sessão de reprodução

```
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: Veja as respostas de sua origem

```
fields @timestamp, responseBody, @message, mediaTailorPath
| filter eventType = "ORIGIN_MANIFEST" and sessionId = "32002de2-837c-4e3e-9660-f3075e8dfd90"
```

Example 3: Visualize o manifesto gerado MediaTailor por para uma determinada sessão

```
fields @timestamp, responseBody, @message
| filter mediaTailorPath like "/v1/master/" and eventType = "GENERATED_MANIFEST" and
sessionId = "32002de2-837c-4e3e-9660-f3075e8dfd90"
```

Example 4: Exibir todos os eventos de um determinado **RequestId**

Use essa consulta para visualizar o manifesto de origem e o manifesto gerado por MediaTailor.

```
fields @timestamp, responseBody, @message, mediaTailorPath
| filter requestId = "e5ba82a5-f8ac-4efb-88a0-55bed21c45b4"
```

Monitoramento AWS Elemental MediaTailor com CloudWatch métricas da Amazon

Você pode monitorar AWS Elemental MediaTailor métricas usando CloudWatch o. CloudWatch coleta dados brutos sobre o desempenho do serviço e processa esses dados em métricas legíveis e quase em tempo real. Essas estatísticas são mantidas por 15 meses, de maneira que você possa acessar informações históricas e ter uma perspectiva melhor de como o aplicativo web ou o serviço está se saindo. Você também pode definir alarmes que observam determinados limites e enviam notificações ou realizam ações quando esses limites são atingidos. Para obter mais informações, consulte o [Guia CloudWatch do usuário da Amazon](#).

As métricas podem ser úteis quando você investiga manifestos obsoletos. Para obter mais informações, consulte [Usando métricas para diagnosticar manifestos obsoletos](#).

As métricas são agrupadas primeiro pelo namespace do serviço e, em seguida, por várias combinações de dimensão dentro de cada namespace.

Para visualizar métricas usando o CloudWatch console

1. Abra o CloudWatch console em <https://console.aws.amazon.com/cloudwatch/>.

2. No painel de navegação, selecione Métricas.
3. Em Todas as métricas, escolha o MediaTailornamespace.
4. Selecione a dimensão da métrica para exibir as métricas (por exemplo, originID).
5. Especifique o período que você deseja visualizar.

Para visualizar métricas usando o AWS Command Line Interface (AWS CLI)

- Em um prompt de comando, use o seguinte comando:

```
aws cloudwatch list-metrics --namespace "AWS/MediaTailor"
```

AWS Elemental MediaTailor CloudWatch métricas

O AWS Elemental MediaTailor namespace inclui as seguintes métricas. Essas métricas são publicadas por padrão na conta.

Métricas de montagem de canais (CA)

Na tabela a seguir, todas as métricas estão disponíveis por canal ou por saída do canal.

Métrica	Description
4xxErrorCount	O número de 4xx erros.
5xxErrorCount	O número de 5xx erros.
RequestCount	O número total de solicitações . A contagem de transações depende muito da frequência com que os jogadores solicitam manifestos atualizados e do número de jogadores. Cada solicitação de player conta como uma transação.
TotalTime	A quantidade de tempo que o servidor do aplicativo levou para processar a solicitação, incluindo o tempo usado para receber bytes e gravar bytes no cliente e na rede.

Server-side Ad-insertion Métricas (SSAI)

A tabela a seguir lista as métricas de inserção de anúncios do lado do servidor.

Métrica	Description
<code>AdDecisionServer.Ads</code>	A contagem de anúncios incluídos nas respostas do servidor de decisão de anúncios (ADS) dentro do CloudWatch período que você especificou. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)
<code>AdDecisionServer.Duration</code>	A duração total, em milissegundos, de todos os anúncios MediaTailor recebidos do ADS dentro do CloudWatch período que você especificou. Essa duração pode ser maior do <code>Avail.Duration</code> que a especificada. Dimensões: <code>ConfigurationName</code>
<code>AdDecisionServer.Errors</code>	O número de respostas de código de status não HTTP 200, respostas vazias e respostas com tempo limite MediaTailor recebidas do ADS dentro do CloudWatch período especificado. Dimensões: <code>ConfigurationName</code>
<code>AdDecisionServer.FillRate</code>	A média simples das taxas em que as respostas do ADS preencheram as disponibilidades de anúncios individuais correspondentes para o período especificado. Para obter a média ponderada, calcule o <code>AdDecisionServer.Duration</code> como

Métrica	Description
	uma porcentagem do <code>Avail.Duration</code> . Para obter mais informações sobre a média ponderada e simples, consulte Médias simples e ponderadas. Dimensões: <code>ConfigurationName</code>
<code>AdDecisionServer.Latency</code>	O tempo de resposta em milissegundos para solicitações feitas pelo MediaTailor ADS. Dimensões: <code>ConfigurationName</code>
<code>AdDecisionServer.Timeouts</code>	O número de solicitações expiradas para o ADS no CloudWatch período que você especificou. Dimensões: <code>ConfigurationName</code>
<code>AdNotReady</code>	O número de vezes que o ADS apontou para um anúncio que ainda não foi transcodificado pelo serviço transcodificador interno no período especificado. Um valor alto dessa métrica pode contribuir para um <code>Avail.FillRate</code> geral baixo. Dimensões: <code>ConfigurationName</code>
<code>AdsBilled</code>	O número de anúncios pelos quais os clientes são MediaTailor cobrados com base na inserção. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)

Métrica	Description
<code>Avail.Duration</code>	O número total planejado de milissegundos do anúncio é disponibilizado dentro do CloudWatch período. O total planejado é baseado nas durações de disponibilidade do anúncio no manifesto de origem. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)
<code>Avail.FilledDuration</code>	O número planejado de milissegundos de tempo de disponibilidade do anúncio que MediaTailor será preenchido com anúncios dentro do CloudWatch período. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)

Métrica	Description
<code>Avail.FillRate</code>	A média simples planejada das taxas nas quais o anúncio individual MediaTailor será preenchido é disponibilizada dentro do período de CloudWatch tempo. Para obter a média ponderada, calcule o <code>Avail.FilledDuration</code> como uma porcentagem do <code>Avail.Duration</code>. Para obter mais informações sobre a média ponderada e simples, consulte Médias simples e ponderadas. O <code>Avail.FillRate</code> máximo que o MediaTailor pode atingir é delimitado pelo <code>AdDecisionServer.FillRate</code>. Se o <code>Avail.FillRate</code> for baixo, compare-o com o <code>AdDecisionServer.FillRate</code>. Se o <code>AdDecisionServer.FillRate</code> for baixo, o ADS pode não estar retornando anúncios o suficiente para as durações de disponibilidades. Dimensões: <code>ConfigurationName</code>

Métrica	Description
<code>Avail.Impression</code>	O número de anúncios que MediaTailor dispararam beacons de impressão durante os relatórios do lado do servidor. MediaTailor emite uma contagem para cada anúncio que tenha pelo menos um URI de impressão , independentemente de esses beacons terem sido finalmente entregues. Use essa métrica como denominador para calcular a taxa de conclusão de vídeo (VCR) (consulte). <code>Avail.Complete</code> Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)
<code>Avail.Complete</code>	O número de anúncios que MediaTailor dispararam beacons completos durante os relatórios do lado do servidor. MediaTailor emite uma contagem para cada anúncio que tenha pelo menos um URI de rastreamento completo, independentemente de esses beacons terem sido finalmente entregues. Para calcular a taxa de conclusão do vídeo (VCR), divida a soma de <code>Avail.Complete</code> pela soma de <code>Avail.Impression</code> Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)

Métrica	Description
<code>Avail.ObservedDuration</code>	O número total observado de milissegundos de disponibilidades de anúncios que ocorreram dentro do CloudWatch período. <code>Avail.ObservedDuration</code> é emitido no final da disponibilidade do anúncio e é baseado na duração dos segmentos informados no manifesto durante a disponibilização do anúncio. Dimensões: <code>ConfigurationName</code>
<code>Avail.ObservedFilledDuration</code>	O número observado de milissegundos de tempo de disponibilidade do anúncio MediaTailor preenchido com anúncios dentro do CloudWatch período. Dimensões: <code>ConfigurationName</code>
<code>Avail.ObservedFillRate</code>	A média simples observada das taxas nas quais o anúncio individual MediaTailor preenchido é disponibilizado dentro do período de CloudWatch tempo. Emitido somente para manifestos HLS, na primeira tag. CUE-IN Se não houver CUE-IN tag, MediaTailor não emite essa métrica. Dimensões: <code>ConfigurationName</code>
<code>Avail.ObservedSlateDuration</code>	O número total observado de milissegundos de ardósia que foi inserido no CloudWatch período. Dimensões: <code>ConfigurationName</code>

Métrica	Description
GetManifest.Age	A idade total do manifesto em milissegundos. Medido desde o momento em que a origem cria o manifesto até o momento em que MediaTailor envia o manifesto personalizado. Para obter mais informações sobre métricas para medir a idade do manifesto, consulte Usando métricas para diagnosticar manifestos obsoletos. Dimensões: ConfigurationName
GetManifest.Errors	O número de erros recebidos durante MediaTailor a geração de manifestos no CloudWatch período que você especificou. Dimensões: ConfigurationName
GetManifest.Latency	O tempo de MediaTailor resposta em milissegundos para a solicitação gerar manifestos. Para obter mais informações sobre métricas para medir a idade do manifesto, consulte Usando métricas para diagnosticar manifestos obsoletos. Dimensões: ConfigurationName

Métrica	Description
GetManifest.MediaTailorAge	A quantidade de tempo em que o manifesto foi armazenado MediaTailor em milissegundos. Medido desde quando MediaTailor recebe uma resposta de origem até quando MediaTailor envia o manifesto personalizado. Para obter mais informações sobre métricas para medir a idade do manifesto, consulte Usando métricas para diagnosticar manifestos obsoletos. Dimensões: ConfigurationName
Origin.Age	A quantidade de tempo em que a origem tem o manifesto em milissegundos. Medido desde o momento em que a origem cria o manifesto até o momento em que MediaTailor envia a solicitação de origem. Todas as <code>origin.*</code> métricas são emitidas para solicitações atendidas diretamente da origem. Eles não são emitidos para respostas de origem em cache. Para obter mais informações sobre métricas para medir a idade do manifesto, consulte Usando métricas para diagnosticar manifestos obsoletos. Dimensões: ConfigurationName

Métrica	Description
<code>Origin.Errors</code>	O número de respostas de código de status não HTTP 200 e respostas com tempo limite MediaTailor recebidas do servidor de origem no CloudWatch período que você especificou. Todas as <code>origin.*</code> métricas são emitidas para solicitações atendidas diretamente da origem. Eles não são emitidos para respostas de origem em cache. Dimensões: <code>ConfigurationName</code>
<code>Origin.ManifestFileSizeBytes</code>	O tamanho do arquivo do manifesto de origem em bytes para HLS e DASH. Normalmente, essa métrica é usada em conjunto com <code>Origin.ManifestFileSizeTooLarge</code>. Todas as <code>origin.*</code> métricas são emitidas para solicitações atendidas diretamente da origem. Eles não são emitidos para respostas de origem em cache. Dimensões: <code>ConfigurationName</code>
<code>Origin.ManifestFileSizeTooLarge</code>	O número de respostas da origem que têm um tamanho de manifesto maior do que o valor configurado. Normalmente, essa métrica é usada em conjunto com <code>Origin.ManifestFileSizeBytes</code>. Todas as <code>origin.*</code> métricas são emitidas para solicitações atendidas diretamente da origem. Eles não são emitidos para respostas de origem em cache. Dimensões: <code>ConfigurationName</code>

Métrica	Description
Origin.Timeouts	O número de solicitações expiradas para o servidor de origem no CloudWatch período que você especificou. Todas as <code>origin.*</code> métricas são emitidas para solicitações atendidas diretamente da origem. Eles não são emitidos para respostas de origem em cache. Dimensões: <code>ConfigurationName</code>
Requests	O número de transações simultâneas por segundo em todos os tipos de solicitação. A contagem de transações depende principalmente do número de jogadores e da frequência com que os jogadores solicitam manifestos atualizados. Cada solicitação de player conta como uma transação. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)

Métrica	Description
<code>SkippedReason.DurationExceeded</code>	O número de anúncios que não foram inseridos em uma disponibilidade porque o ADS retornou uma duração de anúncios maior do que a duração de disponibilidade especificada. Um valor alto dessa métrica pode contribuir para uma discrepância entre as métricas <code>AdsBilled</code> e <code>AdDecisionServer.Ads</code>. Para obter mais informações sobre os motivos pelos quais o anúncio foi ignorado, consulte Solução de problemas com o ignorar anúncios. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)
<code>SkippedReason.EarlyCueIn</code>	O número de anúncios ignorados devido a um anúncio antecipado. CUE-IN Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)
<code>SkippedReason.ImportError</code>	O número de anúncios ignorados devido a um erro no trabalho de importação. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)

Métrica	Description
<code>SkippedReason.ImportInProgress</code>	O número de anúncios ignorados devido a um trabalho de importação ativo existente. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)
<code>SkippedReason.InternalError</code>	O número de anúncios ignorados devido a um erro MediaTailor interno. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)
<code>SkippedReason.NewCreative</code>	O número de anúncios que não foram inseridos em um avail porque foi a primeira vez que o recurso foi solicitado por um cliente. Um valor alto dessa métrica pode contribuir temporariamente para uma baixa <code>generalAvail.FillRate</code>, até que os ativos possam ser transcodificados com sucesso. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)
<code>SkippedReason.NoVariantMatch</code>	O número de anúncios ignorados por não haver correspondência de variantes entre o anúncio e o conteúdo. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)

Métrica	Description
<code>SkippedReason.PersonalizationThresholdExceeded</code>	A duração dos anúncios que excedem a configuração Limite de Personalização nessa configuração. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)
<code>SkippedReason.ProfileNotFound</code>	O número de anúncios ignorados porque o perfil de transcodificação não foi encontrado. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)
<code>SkippedReason.TranscodeError</code>	O número de anúncios ignorados devido a um erro de transcodificação. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)
<code>SkippedReason.TranscodeInProgress</code>	A contagem do número de anúncios que não foram inseridos em um avail porque o anúncio ainda não havia sido transcodificado. Um valor alto dessa métrica pode contribuir temporariamente para uma baixa <code>geralAvail.FillRate</code>, até que os ativos possam ser transcodificados com sucesso. Dimensões: • <code>ConfigurationName</code> • Account-level (sem dimensões)

Note

Nas sessões do HLS Interstitials, algumas métricas se comportam de forma diferente devido à natureza de vinculação tardia da tomada de decisões sobre anúncios:

- `Avail.ObservedFilledDuration` corresponde, `Avail.FilledDuration` pois MediaTailor não consegue observar o comportamento real de reprodução do lado do cliente.
- `Avail.ObservedSlateDuration` relata a duração planejada da lista a partir das respostas da Lista de ativos, em vez da reprodução observada.
- As métricas prefixadas com “Observado” fornecem valores estimados para as sessões do HLS Interstitials.

Métricas de funções

As métricas a seguir são publicadas quando você usa o [Functions](#) com suas configurações de reprodução. Essas métricas são sempre emitidas quando uma função é configurada em um gancho de ciclo de vida. Nenhuma configuração de inscrição ou registro é necessária.

Hook-level métricas — um ponto de dados por execução do gancho do ciclo de vida:

Métrica	Description
<code>PreSessionInitHook.Invocations</code>	O número de vezes que o gancho de inicialização pré-sessão foi invocado. Dimensões: <code>ConfigurationName</code>
<code>PreSessionInitHook.Errors</code>	O número de execuções de ganchos de inicialização pré-sessão que resultaram em um erro. Dimensões: <code>ConfigurationName</code>
<code>PreSessionInitHook.Latency</code>	O tempo de execução em milissegundos do gancho de inicialização da pré-sessão. Dimensões: <code>ConfigurationName</code>

Métrica	Description
<code>PreAdsRequestHook.Invocations</code>	O número de vezes que o gancho de solicitação de pré-anúncios foi invocado. Dimensões: <code>ConfigurationName</code>
<code>PreAdsRequestHook.Errors</code>	O número de execuções de ganchos de solicitações pré-anúncios que resultaram em um erro. Dimensões: <code>ConfigurationName</code>
<code>PreAdsRequestHook.Latency</code>	O tempo de execução em milissegundos para o gancho de solicitação de pré-anúncios. Dimensões: <code>ConfigurationName</code>

Function-level métricas — um ponto de dados por execução de função individual. Essas métricas incluem dimensões adicionais (`FunctionId`, `FunctionType`, `HookType`) para que você possa identificar qual função específica está lenta ou está falhando:

Métrica	Description
<code>Function.Invocations</code>	O número de vezes que uma função individual foi executada. Dimensões: <code>ConfigurationName</code> <code>ConfigurationName</code> , <code>FunctionId</code> , <code>FunctionType</code> , <code>HookType</code>
<code>Function.Errors</code>	O número de execuções de funções individuais que resultaram em um erro. Dimensões: <code>ConfigurationName</code>

Métrica	Description
	• <code>ConfigurationName</code> , <code>FunctionId</code> , <code>FunctionType</code> , <code>HookType</code>
<code>Function.Latency</code>	O tempo de execução em milissegundos para uma função individual. Dimensões: • <code>ConfigurationName</code> • <code>ConfigurationName</code> , <code>FunctionId</code> , <code>FunctionType</code> , <code>HookType</code>

Médias simples e ponderadas

Você pode recuperar a média simples e a média ponderada das respostas do ADS às solicitações de anúncios MediaTailor e de como os MediaTailor preenchimentos são disponibilizados:

- As médias simples são fornecidas no `AdDecisionServer.FillRate` e no `Avail.FillRate`. Essas são as médias da taxa de preenchimento para invocar porcentagens para disponibilidades individuais para o período. A média simples não leva em conta as diferenças entre as durações das disponibilidades individuais.
- As médias ponderadas são as porcentagens da taxa de preenchimento para a soma de todas as durações de disponibilidades. Elas são calculadas como $(\text{AdDecisionServer.Duration} * 100) / \text{Avail.Duration}$ e $(\text{Avail.FilledDuration} * 100) / \text{Avail.Duration}$. Essas médias refletem as diferenças na duração de cada disponibilidade de anúncios, dando mais peso para aquelas com maior duração.

Para um período que contém apenas uma única disponibilidade de anúncios, a média simples fornecida pelo `AdDecisionServer.FillRate` é igual à média ponderada fornecida pelo $(\text{AdDecisionServer.Duration} * 100) / \text{Avail.Duration}$. A média simples fornecida pelo `Avail.FillRate` é igual à média ponderada fornecida pelo $(\text{Avail.FilledDuration} * 100) / \text{Avail.Duration}$.

Exemplo

Vamos supor que o período que você especificou tenha as duas disponibilidades de anúncios a seguir:

- A primeira disponibilidade de anúncios tem 90 segundos de duração:
 - A resposta ADS para a disponibilidade fornece 45 segundos de anúncios (50% preenchido).
 - MediaTailor preenche 45 segundos do tempo de anúncio disponível (50% preenchido).
- A segunda disponibilidade de anúncios tem 120 segundos de duração:
 - A resposta ADS para a disponibilidade fornece 120 segundos de anúncios (100% preenchido).
 - MediaTailor preenche 90 segundos do tempo de anúncio disponível (75% preenchido).

As métricas são as seguintes:

- `Avail.Duration` é 210, a soma das duas durações de disponibilidade de anúncios: $90 + 120$.
- `AdDecisionServer.Duration` é 165, a soma das duas durações de resposta: $45 + 120$.
- `Avail.FilledDuration` é 135, a soma das duas durações preenchidas: $45 + 90$.
- `AdDecisionServer.FillRate` é 75%, a média das porcentagens preenchidas para cada disponibilidade: $(50\% + 100\%)/2$. Essa é a média simples.
- A média ponderada para as taxas de preenchimento ADS é 78,57%, que é `AdDecisionServer.Duration` como uma porcentagem do `Avail.Duration`: $(165 \times 100)/210$. Esse cálculo leva em conta as diferenças nas durações.
- `Avail.FillRate` é 62,5%, a média das porcentagens preenchidas para cada disponibilidade: $(50\% + 75\%)/2$. Essa é a média simples.
- A média ponderada das taxas de preenchimento MediaTailor disponível é de 64,29%, que é a porcentagem de `Avail.Duration`: $(Avail.FilledDuration 135 \times 100)/210$. Esse cálculo leva em conta as diferenças nas durações.

O máximo `Avail.FillRate` que se MediaTailor pode atingir com a disponibilidade de qualquer anúncio é 100%. O ADS pode retornar mais tempo de anúncios do que está disponível na disponibilidade, mas o MediaTailor só pode preencher o tempo disponível.

AWS Elemental MediaTailor CloudWatch dimensões

Você pode filtrar os AWS Elemental MediaTailor dados usando as seguintes dimensões.

Dimensão	Description
ConfigurationName	Indica a configuração à qual a métrica pertence. Disponível em todas as métricas.
FunctionId	O identificador da função. Disponível somente em <code>Function.*</code> métricas.
FunctionType	O tipo de função: <code>CUSTOM_OUTPUT</code> , <code>HTTP_REQUEST</code> , <code>SEQUENTIAL_EXECUTOR</code> , ou <code>CONCURRENT_EXECUTOR</code> . Disponível somente em <code>Function.*</code> métricas.
HookType	O gancho do ciclo de vida que acionou a função: <code>PRE_SESSION_INITIALIZATION</code> ou <code>PRE_ADS_REQUEST</code> . Disponível somente em <code>Function.*</code> métricas.

Usando métricas para diagnosticar manifestos obsoletos de AWS Elemental MediaTailor

Um manifesto obsoleto é aquele que não foi atualizado recentemente. Diferentes fluxos de trabalho de inserção de anúncios podem ter uma tolerância variável quanto ao tempo necessário para que um manifesto seja considerado obsoleto, com base em vários fatores (como requisitos para sistemas posteriores). Você pode usar CloudWatch as métricas da Amazon para identificar manifestos que excedem a tolerância de obsolescência do seu fluxo de trabalho e ajudar a identificar o que pode estar causando os atrasos nas atualizações dos manifestos.

As métricas a seguir ajudam a identificar manifestos obsoletos e suas causas.

Para obter informações sobre todas as métricas MediaTailor emitidas, consulte [AWS Elemental MediaTailor CloudWatch métricas](#).

Métrica	Definição	Use
<code>GetManifest.Age</code>	Mede a idade total do manifesto, incluindo ambos <code>GetManifest.MediaTailorAge</code> e <code>Origin.Age</code> para essa configuração.	Você pode usar essa métrica para identificar manifestos que ultrapassem seu limite de atualização e estão obsoletos. Defina alarmes nessa métrica para que você seja alertado quando manifestos obsoletos estiverem sendo veiculados. Para obter informações sobre alarmes, consulte Alarmes sobre métricas no Guia CloudWatch do usuário da Amazon. Ao receber um alarme, use <code>Origin.Age</code> e <code>GetManifest.MediaTailorAge</code> para identificar se MediaTailor ou a origem está causando a obsolescência.
<code>Origin.Age</code>	Mede por quanto tempo a origem tem o manifesto antes de enviá-lo MediaTailor para essa configuração. Essa métrica não é emitida quando a resposta vem de uma rede de distribuição de conteúdo (CDN). A resposta deve vir da origem <code>Origin.Age</code> para ser emitida.	Ao identificar manifestos obsoletos com <code>GetManifest.Age</code>, você pode analisar a <code>Origin.Age</code> métrica e a <code>GetManifest.MediaTailorAge</code> métrica para determinar o que está contribuindo para a obsolescência manifesta. Se você achar que <code>Origin.Age</code> é mais longo do que o tempo normal de processamento na origem, é provável que o sistema upstream esteja causando o problema e você deve focar o diagnóstico nele.

Métrica	Definição	Use
<code>GetManifest.MediaTailorAge</code>	Mede por quanto tempo esse manifesto MediaTailor foi armazenado para essa configuração.	Ao identificar manifestos obsoletos com <code>GetManifest.Age</code>, você pode analisar a <code>GetManifest.MediaTailorAge</code> métrica e a <code>Origin.Age</code> métrica para determinar o que está contribuindo para a obsolescência manifesta. Se o tempo de personalização do manifesto <code>GetManifest.MediaTailorAge</code> for maior do que o tempo normal de personalização MediaTailor, é provável que isso MediaTailor esteja causando o problema, e você deve focar nos diagnósticos. <code>GetManifest.Latency</code> pode identificar ainda mais quanto tempo é necessário MediaTailor para criar um manifesto personalizado.

Métrica	Definição	Use
GetManifest.Latency	Mede o tempo necessário para processar MediaTailor a solicitação e criar um manifesto personalizado para essa configuração.	Ao comparar <code>Origin.Age</code> <code>GetManifest.MediaTailorAge</code> e determinar qual MediaTailor é a causa do atraso na entrega do manifesto, você pode analisar a <code>GetManifest.Latency</code> métrica para determinar se o processo de personalização do manifesto está contribuindo para a obsolescência do manifesto. <code>GetManifest.MediaTailorAge</code> mede o tempo total em que o manifesto é armazenado MediaTailor. <code>GetManifest.Latency</code> mede quanto tempo de armazenamento é usado para MediaTailor personalizar o manifesto em resposta a uma solicitação.

Gravação AWS Elemental MediaTailor Chamadas de API

AWS Elemental MediaTailor é integrado com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço em MediaTailor. CloudTrail captura todas as chamadas de API MediaTailor como eventos. As chamadas capturadas incluem chamadas do MediaTailor console e chamadas de código para as operações MediaTailor da API. Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para MediaTailor. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos. Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita MediaTailor, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Para saber mais sobre isso CloudTrail, consulte o [Guia AWS CloudTrail do usuário](#).

AWS Elemental MediaTailor informações em CloudTrail

CloudTrail é ativado em sua AWS conta quando você cria a conta. Quando a atividade ocorre em AWS Elemental MediaTailor, essa atividade é registrada em um CloudTrail evento junto com outros eventos AWS de serviço no histórico de eventos. Você pode visualizar, pesquisar e baixar eventos recentes em sua AWS conta. Para obter mais informações, consulte [Visualização de eventos com histórico de CloudTrail eventos](#).

Para um registro contínuo dos eventos em sua AWS conta, incluindo eventos para AWS Elemental MediaTailor, crie uma trilha. Uma trilha permite CloudTrail entregar arquivos de log para um bucket do Amazon S3. Por padrão, quando você cria uma trilha no console, a trilha se aplica a todas as AWS regiões. A trilha registra eventos de todas as regiões na AWS partição e entrega os arquivos de log ao bucket do Amazon S3 que você especificar. Além disso, você pode configurar outros AWS serviços para analisar e agir com base nos dados de eventos coletados nos CloudTrail registros. Para saber mais, consulte:

- [Criando uma trilha para sua AWS conta](#)
- [AWS integrações de serviços com registros CloudTrail](#)
- [Configurando notificações do Amazon SNS para CloudTrail](#)
- [Recebendo arquivos de CloudTrail log de várias regiões](#) e [Recebendo arquivos de CloudTrail log de várias contas](#)

Todas AWS Elemental MediaTailor as ações são registradas CloudTrail e documentadas na [referência da AWS Elemental MediaTailor API](#). Por exemplo, as chamadas para as operações `PutPlaybackConfiguration` e `ListPlaybackConfigurations` geram entradas nos arquivos de log do CloudTrail.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar o seguinte:

- Se a solicitação foi feita com as credenciais do usuário raiz ou AWS Identity and Access Management (IAM)
- Se a solicitação foi feita com credenciais de segurança temporárias de um perfil ou de um usuário federado
- Se a solicitação foi feita por outro AWS serviço

Para obter mais informações, consulte [Elemento userIdentity do CloudTrail](#).

Noções básicas AWS Elemental MediaTailor entradas do arquivo de log

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3 que você especificar. CloudTrail os arquivos de log contêm uma ou mais entradas de log. Um evento representa uma única solicitação de qualquer fonte e inclui informações sobre a ação solicitada, a data e hora da ação, parâmetros de solicitação, e assim por diante. arquivos de log do CloudTrail não são um rastreamento de pilha ordenada das chamadas da API pública. Assim, elas não são exibidas em nenhuma ordem específica.

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a PutPlaybackConfiguration ação:

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDAEXAMPLE",
    "arn": "arn:aws:iam::111122223333:user/testuser",
    "accountId": "111122223333",
    "accessKeyId": "AIDAEXAMPLE",
    "userName": "testuser"
  },
  "eventTime": "2018-12-28T22:53:46Z",
  "eventSource": "mediatailor.amazonaws.com",
  "eventName": "PutPlaybackConfiguration",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "1.2.3.4",
  "userAgent": "PostmanRuntime/7.4.0",
  "requestParameters": {
    "VideoContentSourceUrl": "http://examplevideo.com",
    "Name": "examplename",
    "AdDecisionServerUrl": "http://exampleads.com"
  },
  "responseElements": {
    "SessionInitializationEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/session/AKIAIOSFODNN7EXAMPLE/examplename/",
    "DashConfiguration": {
      "ManifestEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/dash/AKIAIOSFODNN7EXAMPLE/examplename/",
      "MpdLocation": "EMT_DEFAULT"
    },
    "AdDecisionServerUrl": "http://exampleads.com",
```

```

    "CdnConfiguration": {},
    "PlaybackEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com",
    "HlsConfiguration": {
        "ManifestEndpointPrefix": "https://777788889999.mediatailor.us-east-1.amazonaws.com/v1/master/AKIAIOSFODNN7EXAMPLE/exemplename/"
    },
    "VideoContentSourceUrl": "http://examplevideo.com",
    "Name": "exemplename"
},
"requestID": "1a2b3c4d-1234-5678-1234-1a2b3c4d5e6f",
"eventID": "987abc65-1a2b-3c4d-5d6e-987abc654def",
"readOnly": false,
"eventType": "AwsApiCall",
"recipientAccountId": "111122223333"
}

```

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a `GetPlaybackConfiguration` ação:

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDAEXAMPLE",
    "arn": "arn:aws:iam::111122223333:user/testuser",
    "accountId": "111122223333",
    "accessKeyId": "AIDAEXAMPLE",
    "userName": "testuser"
  },
  "eventTime": "2018-12-28T22:52:37Z",
  "eventSource": "mediatailor.amazonaws.com",
  "eventName": "GetPlaybackConfiguration",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "1.2.3.4",
  "userAgent": "PostmanRuntime/7.4.0",
  "requestParameters": {
    "Name": "exemplename"
  },
  "responseElements": null,
  "requestID": "0z1y2x3w-0123-4567-9876-6q7r8s9t0u1v",
  "eventID": "888ddd77-3322-eeww-uuii-abc123jkl343",
  "readOnly": true,

```



```

    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
  }

```

Recebendo AWS Elemental MediaTailor alertas de montagem de canais

MediaTailor cria alertas para problemas ou possíveis problemas que ocorram com seus recursos de montagem de canais. O alerta descreve o problema, quando o problema ocorreu e os recursos afetados.

Você pode visualizar os alertas no Console de gerenciamento da AWS, no AWS Command Line Interface (AWS CLI), nos AWS SDKs ou programaticamente usando a API. MediaTailor [ListAlerts](#)

Important

Os alertas estão disponíveis somente para recursos de montagem de canais criados em ou após 14 de julho de 2021.

Alertas de montagem de canais

Tipo de alerta	Código de alerta	Mensagem de alerta	Observações
Fonte VOD	NOT_PROCESSED	MediaTailor não processou a configuração do pacote <i>configurationPath</i> .	
	UNREACHABLE	Não conseguimos acessar o URL <i>url</i> .	
	UNAUTHORIZED	<i>url</i> não autorizou a solicitação.	
	TIMEOUT	A conexão com o <i>url</i> atingiu o tempo limite.	

Tipo de alerta	Código de alerta	Mensagem de alerta	Observações
	UNPARSABLE_MANIFEST	MediaTailor encontrou um problema ao analisar o manifesto de <i>url</i> .	
	VARIANT_DURATION_MISMATCH	MediaTailor encontrou variantes com durações totais incompatíveis ao analisar o manifesto de <i>url</i> . Isso pode causar paralisação durante a reprodução.	Seu manifesto tem durações variadas. variants/representations Isso pode resultar em legendas ausentes ou incorretas e na MediaTailor impossibilidade de inserir anúncios.
	SEGMENT_DURATION_TOO_LONG	MediaTailor encontrou um segmento com uma duração maior que trinta segundos ao analisar o manifesto de <i>url</i> . Isso pode causar paralisação durante a reprodução, legendas ausentes ou incorretas e a incapacidade de inserir anúncios.	Seu manifesto contém um segmento com mais de 30 segundos.
	TARGET_DURATION_MISMATCH	MediaTailor encontrou uma incompatibilidade de EXT-X-TARGETDURATION valores nos manifestos do HLS ao analisar o manifesto de <i>url</i> . Isso pode causar paralisação durante a reprodução.	A duração alvo não coincide com todos os manifestos na origem.
Localização da fonte	NOT_PROCESSED	MediaTailor não processou o recurso <i>resourceName</i> .	

Tipo de alerta	Código de alerta	Mensagem de alerta	Observações
Programa	VOD_SOURCE_ALERT	A fonte de VOD <i>vodSourceName</i> neste programa tem o seguinte alerta: <i>vodSourceAlertCode</i> <i>vodSourceAlertMessage</i>	
	SOURCE_LOCATION_ALERT	O local de origem <i>sourceLocationName</i> contido neste programa tem o seguinte alerta <i>sourceLocationAlertCode</i> : <i>sourceLocationAlertMessage</i>	
	CODEC_MISMATCH	MediaTailor encontrou um codec incompatível no cronograma. <i>channelName</i> A incompatibilidade está <i>sourceGroupName</i> entre o <i>programName1</i> manifesto <i>manifestUrl</i> e o <i>programName2</i> <i>manifestUrl</i> manifesto.	
	RESOLUTION_MISMATCH	MediaTailor encontrou uma resolução incompatível no <i>channelName</i> cronograma. A incompatibilidade está <i>sourceGroupName</i> entre o <i>programName1</i> manifesto <i>manifestUrl</i> e o <i>programName2</i> <i>manifestUrl</i> manifesto.	

Tipo de alerta	Código de alerta	Mensagem de alerta	Observações
	BANDWIDTH_MISMATCH	MediaTailor encontrou largura de banda incompatível no cronograma. <i>channelName</i> A incompatibilidade está <i>sourceGroupName</i> entre o <i>programName1</i> manifesto <i>manifestUrl</i> e o <i>programName2 manifestUrl</i> manifesto.	
	FRAMERATE_MISMATCH	MediaTailor encontrou uma taxa de quadros incompatível no cronograma. <i>channelName</i> A incompatibilidade está <i>sourceGroupName</i> entre o <i>programName1</i> manifesto <i>manifestUrl</i> e o <i>programName2 manifestUrl</i> manifesto.	
	TARGET_DURATION_MISMATCH	MediaTailor encontrou EXT-X-TARGETDURATION valores incompatíveis nos manifestos HLS no cronograma. <i>channelName</i> A incompatibilidade está <i>sourceGroupName</i> entre o <i>programName1</i> manifesto <i>manifestUrl</i> e o <i>programName2 manifestUrl</i> manifesto.	

Tipo de alerta	Código de alerta	Mensagem de alerta	Observações
	SEGMENT_DURATION_MISMATCH	MediaTailor encontrou valores de duração de segmentos incompatíveis em todos os manifestos no cronograma. <i>channelName</i> A incompatibilidade está <i>sourceGroup</i> entre o <i>programName1</i> manifesto <i>manifestUrl</i> e o <i>programName2</i> manifesto <i>manifestUrl</i> manifesto.	
	NO_COMMON_SEGMENT_BOUNDARY_FOR_AD_SLATE	MediaTailor Não conseguiu inserir uma lista de anúncios no offset <i>offsetMillis</i> do programa. <i>programName</i> Não há limite de segmento comum no horário de início do painel de anúncios.	
	NOT_PROCESSED	MediaTailor não processou o recurso <i>resourceName</i> .	
	TOO_MANY_ALERTS	MediaTailor encontrou muitos alertas e não fornecerá mais alertas para <i>programName</i> . Limpe os alertas existentes para continuar recebendo alertas <i>programName</i> .	
Canal	PROGRAM_ALERT	O programa <i>programName</i> contido neste canal tem o seguinte alerta <i>programAlertCode</i> : <i>programAlertMessage</i>	

Solução de problemas de alertas relacionados ao manifest

Ao encontrar alertas relacionados ao manifesto `UNPARSABLE_MANIFEST`, como `VARIANT_DURATION_MISMATCH`, ou `TARGET_DURATION_MISMATCH`, verifique o seguinte:

1. Verifique os requisitos de origem: certifique-se de que seu conteúdo atenda aos requisitos descritos em [the section called “Integrando uma fonte de conteúdo”](#).
2. Verifique os registros do manifesto: para obter uma análise detalhada do erro manifesto e os tipos de eventos, consulte [the section called “Registros de manifestos”](#).
3. Verifique os códigos de erro: para obter uma análise detalhada do erro, consulte [Solução de problemas](#).
4. Revise o formato do manifesto: verifique se o manifesto segue as especificações de formato corretas para seu protocolo de streaming (HLS ou DASH).

Visualizando alertas

Você pode visualizar alertas para qualquer recurso de montagem de MediaTailor canais. Quando você visualiza os alertas de canais e programas, MediaTailor inclui todos os recursos relacionados contidos no canal ou programa. Por exemplo, ao visualizar os alertas de um programa específico, você também vê alertas para a localização da fonte e as fontes de VOD que o programa contém.

Para visualizar alertas, execute o procedimento a seguir.

Console

Para visualizar alertas no console

1. Abra o MediaTailor console em <https://console.aws.amazon.com/mediatailor/>.
2. Escolha o recurso para o qual você deseja ver os alertas.
3. Selecione a guia Alertas para ver os alertas.

AWS Command Line Interface (AWS CLI)

Para listar alertas para um recurso de montagem de canais, você precisa do [Amazon Resource Name \(ARN\) do recurso](#). Você pode usar o `describe-resource_type` comando no AWS Command Line Interface (AWS CLI) para obter o ARN do recurso. Por exemplo, execute o comando [describe-channel](#) para obter o ARN de um canal específico:

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

Em seguida, use o comando [aws mediatailor list-alerts](#) para listar os alertas associados ao recurso:

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

API

Para listar alertas para um recurso de montagem de canais, você precisa do [Amazon Resource Name \(ARN\) do recurso](#). Você pode usar a `DescribeResource` operação na MediaTailor API para obter o ARN do recurso. Por exemplo, use a [DescribeChannel](#) operação para obter o ARN de um canal específico.

Em seguida, use a [ListAlerts](#) API para listar os alertas do recurso.

Gerenciamento de alertas

Quando ocorrer um alerta, visualize os alertas no AWS Command Line Interface (AWS CLI), AWS SDKs ou a API de MediaTailor alertas para determinar as possíveis fontes do problema. Console de gerenciamento da AWS

Depois de resolver o problema, MediaTailor apaga o alerta.

Tags AWS Elemental MediaTailor recursos

Uma tag é um rótulo de metadados que você atribui ou AWS atribui a um AWS recurso. Cada tag consiste em uma chave e um valor. Em tags atribuídas por você, você mesmo define a chave e o valor. Por exemplo, você pode definir a chave como `stage` e o valor de um atributo como `test`.

As tags ajudam a:

- Identifique e organize seus AWS recursos. Muitos AWS serviços oferecem suporte à marcação, então você pode atribuir a mesma tag a recursos de serviços diferentes para indicar que os recursos estão relacionados. Por exemplo, você pode atribuir a mesma tag a um AWS Elemental MediaPackage canal e endpoint que você atribui a uma AWS Elemental MediaTailor configuração.

- Acompanhe seus AWS custos. Você ativa essas tags no Gerenciamento de Faturamento e Custos da AWS painel. AWS usa as tags para categorizar seus custos e entregar um relatório mensal de alocação de custos para você. Para obter mais informações, consulte [Usar etiquetas de alocação de custos](#) no [Guia do Usuário do AWS Billing](#).
- Controle o acesso aos seus AWS recursos. Para mais informações, consulte [Controlar o acesso usando etiquetas](#) no [Guia do usuário do IAM](#).

As seções a seguir fornecem mais informações sobre tags para AWS Elemental MediaTailor.

Recursos suportados em AWS Elemental MediaTailor

Os seguintes recursos em oferecem AWS Elemental MediaTailor suporte à marcação:

- Canais
- Configurações
- SourceLocations
- VodSources

Restrições de tag

As seguintes restrições básicas se aplicam às tags nos AWS Elemental MediaTailor recursos:

- Número máximo de tags que você pode atribuir a um recurso: 50
- Comprimento máximo da chave: 128 caracteres Unicode
- Comprimento máximo do valor: 256 caracteres Unicode
- Caracteres válidos para chave e valor — a-z A-Z, 0-9, espaço e os seguintes caracteres: _ . / = + - e @
- Chaves e valores diferenciam maiúsculas de minúsculas
- Não use `aws :` como prefixo para chaves, pois ele é reservado para uso da AWS

Gerenciando tags em AWS Elemental MediaTailor

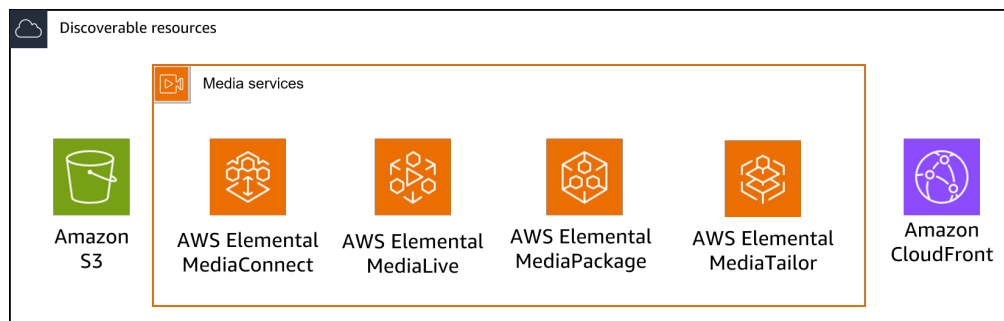
Você define as tags como propriedades em um recurso. Você pode adicionar, editar e excluir tags por meio da AWS Elemental MediaTailor API ou do AWS Command Line Interface (AWS CLI). Para obter mais informações, consulte a [Referência da API AWS Elemental MediaTailor](#).

Monitoramento AWS serviços de mídia com monitor de fluxo de trabalho

O monitor de fluxo de trabalho é uma ferramenta para a descoberta, visualização e monitoramento de fluxos de trabalho de AWS mídia. O monitor de fluxo de trabalho está disponível no AWS console e na API. É possível usar o monitor de fluxo de trabalho para descobrir e criar mapeamentos visuais dos recursos do seu fluxo de trabalho, denominados mapas de sinais. Você pode criar e gerenciar modelos de CloudWatch alarme e EventBridge regras da Amazon para monitorar os recursos mapeados. Os modelos de monitoramento que você cria são transformados em AWS CloudFormation modelos implantáveis para permitir a repetibilidade. AWS os modelos de alarme recomendados fornecem monitoramento predefinido de melhores práticas.

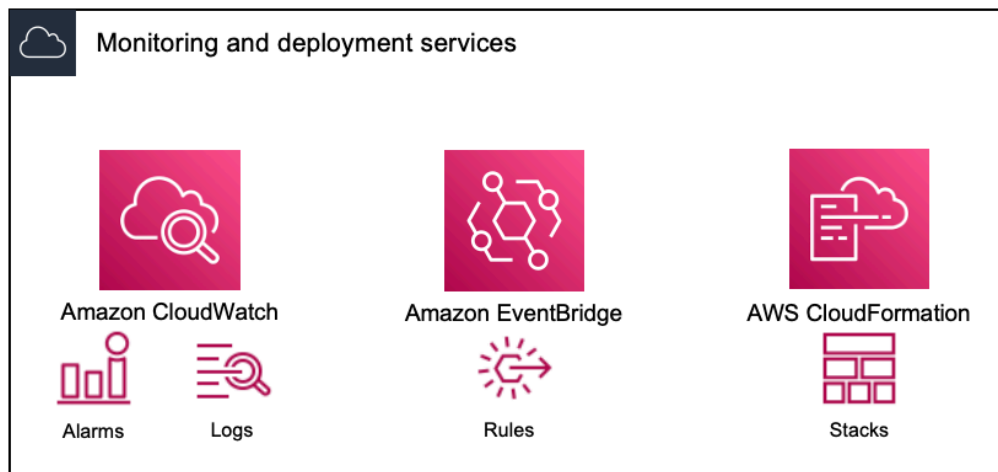
Descobrir

Utilize mapas de sinal para descobrir automaticamente os AWS recursos interconectados associados ao seu fluxo de trabalho de mídia. A descoberta pode ter início em qualquer recurso de serviço compatível e criar um mapeamento completo do fluxo de trabalho. Os mapas de sinais podem ser usados como ferramentas de visualização independentes ou aprimorados com modelos de monitoramento.



Monitorar

Você pode criar modelos personalizados de CloudWatch alarmes e EventBridge regras para monitorar a integridade e o status de seus fluxos de trabalho de mídia. Modelos de alarme de práticas recomendadas estão disponíveis para importação em seu ambiente de monitoramento de fluxo de trabalho. É possível usar os modelos de alarme de práticas recomendadas da forma como estão ou editá-los para se adaptarem melhor ao seu fluxo de trabalho. Todos os modelos criados são transformados em modelos do CloudFormation para implantação repetível.



Note

Não há custo direto para usar o monitor de fluxo de trabalho. No entanto, há custos associados aos recursos criados e usados para monitorar o fluxo de trabalho.

Quando o monitoramento é implantado, a Amazon CloudWatch e EventBridge os recursos da Amazon são criados. Ao usar o AWS Management Console, antes de implantar o monitoramento em um mapa de sinais, você será notificado sobre quantos recursos serão criados. Para obter mais informações sobre preços, consulte: [CloudWatchpreços](#) e [EventBridge preços](#).

O monitor de fluxo de trabalho usa AWS CloudFormation modelos para implantar os EventBridge recursos CloudWatch e. Esses modelos são armazenados em um bucket de classe padrão do Amazon Simple Storage Service, que é criado em seu nome pelo monitor de fluxo de trabalho durante o processo de implantação e gera cobranças de armazenamento e recall de objetos. Para ter mais informações sobre a definição de preços, consulte [Definição de preço do Amazon S3](#).

As visualizações geradas no mapa de sinal do monitor de fluxo de trabalho para AWS Elemental MediaPackage canais são fornecidas pelo MediaPackage Origin Endpoint e incorrerão em taxas de transferência de dados para fora. Para saber os preços, consulte: [MediaPackagepreços](#).

Componentes do monitor de fluxo de trabalho

O monitor de fluxo de trabalho tem quatro componentes principais:

- CloudWatch modelos de alarme - Defina as condições que você gostaria de monitorar usando CloudWatch. Você pode criar seus próprios modelos de alarme ou importar modelos predefinidos criados por AWS. Para obter mais informações, consulte: [CloudWatch grupos de alarmes e modelos para monitorar seu AWS fluxo de trabalho de mídia](#)
- EventBridge modelos de regras - Defina como EventBridge envia notificações quando um alarme é acionado. Para obter mais informações, consulte: [EventBridge grupos de regras e modelos para monitorar seu AWS fluxo de trabalho de mídia](#)
- Mapas de sinais - Use um processo automatizado para criar mapas de fluxo de trabalho AWS elementares usando AWS os recursos existentes. Os mapas de sinais podem ser usados para descobrir recursos em seu fluxo de trabalho e implantar monitoramento nesses recursos. Para ter mais informações, consulte [Mapas de sinais do monitor de fluxo de trabalho](#).
- Visão geral: a página de visão geral permite monitorar diretamente o status de vários mapas de sinais em um único local. Analise métricas, logs e alarmes dos fluxos de trabalho. Para ter mais informações, consulte [Visão geral do monitor de fluxo de trabalho](#).

Serviços com suporte

O monitor de fluxo de trabalho é compatível com a descoberta automática e o mapeamento de sinais de recursos associados aos seguintes serviços:

- AWS Elemental MediaConnect
- AWS Elemental MediaLive
- AWS Elemental MediaPackage
- AWS Elemental MediaTailor
- Amazon S3
- Amazon CloudFront

Tópicos

- [Configurando o monitor de fluxo de trabalho para monitorar AWS serviços de mídia](#)
- [Usar o monitor de fluxo de trabalho](#)

Configurando o monitor de fluxo de trabalho para monitorar AWS serviços de mídia

Para configurar o monitor de fluxo de trabalho pela primeira vez, é necessário criar os modelos de alarme e de eventos e descobrir mapas de sinais que são usados para monitorar fluxos de trabalho de mídia. O guia a seguir contém as etapas necessárias para configurar os perfis do IAM em nível de administrador e de operador, criar recursos de monitor do fluxo de trabalho e implantar o monitoramento nos fluxos de trabalho.

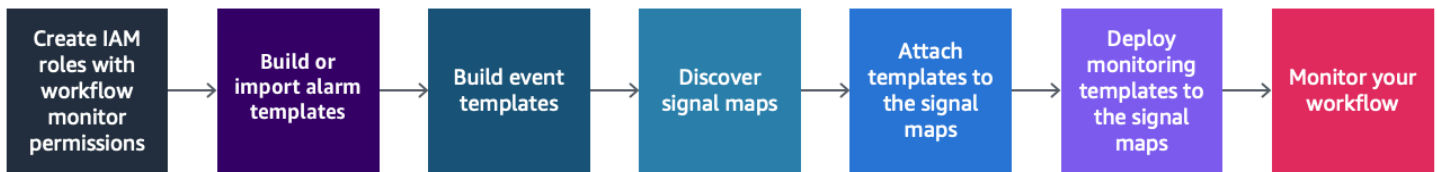
Tópicos

- [Conceitos básicos do monitor de fluxo de trabalho](#)
- [Grupos e modelos do monitor de fluxo de trabalho](#)
- [Mapas de sinais do monitor de fluxo de trabalho](#)
- [Cotas do monitor de fluxo de trabalho](#)

Conceitos básicos do monitor de fluxo de trabalho

As etapas a seguir apresentam uma visão geral básica do primeiro uso do monitor de fluxo de trabalho.

1. Configurar as permissões do IAM do monitor de fluxo de trabalho para perfis em nível de administrador e de operador: [Políticas do IAM do monitor de fluxo de trabalho](#)
2. Crie modelos de alarme ou importe modelos predefinidos criados por: AWS [Alarmes do CloudWatch](#)
3. Crie eventos de notificação que serão entregues por EventBridge: [EventBridge regras](#)
4. Descubra mapas de sinais usando seus recursos AWS elementais existentes: [Mapas de sinais](#)
5. Anexar os modelos de alarme e as regras de notificação ao mapa de sinais: [Anexar modelos](#)
6. Implantar os modelos para começar a monitorar o mapa de sinais: [Implantar modelos de monitoramento](#)
7. Monitorar e analisar recursos do monitor de fluxo de trabalho usando a seção de visão geral do Console da AWS : [Visão geral do](#)



Políticas do IAM do monitor de fluxo de trabalho

O monitor de fluxo de trabalho interage com vários AWS serviços para criar mapas de sinais, compilações, EventBridge recursos CloudWatch e CloudFormation modelos. Como o monitor de fluxo de trabalho interage com uma ampla variedade de serviços, políticas específicas AWS Identity and Access Management (IAM) devem ser atribuídas a esses serviços. Os exemplos a seguir indicam as políticas do IAM necessárias para os perfis do IAM de administrador e de operador.

Política do IAM de administrador

O exemplo de política a seguir destina-se a uma política do IAM do monitor de fluxo de trabalho em nível de administrador. Esse perfil possibilita a criação e o gerenciamento de recursos do monitor de fluxo de trabalho e dos recursos de serviço compatíveis que interagem com o monitor de fluxo de trabalho.

JSON

```
{

  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:List*",
        "cloudwatch:Describe*",
        "cloudwatch:Get*",
        "cloudwatch:PutAnomalyDetector",
        "cloudwatch:PutMetricData",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch:PutCompositeAlarm",
        "cloudwatch:PutDashboard",
        "cloudwatch>DeleteAlarms",
        "cloudwatch>DeleteAnomalyDetector",
        "cloudwatch>DeleteDashboards",
        "cloudwatch:TagResource",
```

```
    "cloudwatch:UntagResource"
  ],
  "Resource": "*"
},
{
  "Effect": "Allow",
  "Action": [
    "cloudformation:List*",
    "cloudformation:Describe*",
    "cloudformation:CreateStack",
    "cloudformation:UpdateStack",
    "cloudformation>DeleteStack",
    "cloudformation:TagResource",
    "cloudformation:UntagResource"
  ],
  "Resource": "*"
},
{
  "Effect": "Allow",
  "Action": [
    "cloudfront:List*",
    "cloudfront:Get*"
  ],
  "Resource": "*"
},
{
  "Effect": "Allow",
  "Action": [
    "ec2:DescribeNetworkInterfaces"
  ],
  "Resource": "*"
},
{
  "Effect": "Allow",
  "Action": [
    "events:List*",
    "events:Describe*",
    "events:CreateEventBus",
    "events:PutRule",
    "events:PutTargets",
    "events:EnableRule",
    "events:DisableRule",
    "events>DeleteRule",
    "events:RemoveTargets",
```

```
        "events:TagResource",
        "events:UntagResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "logs:Describe*",
        "logs:Get*",
        "logs:TagLogGroup",
        "logs:TagResource",
        "logs:UntagLogGroup",
        "logs:UntagResource"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediaconnect:List*",
        "mediaconnect:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "medialive:*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackage:List*",
        "mediapackage:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackagev2:List*",
```

```
        "mediapackagev2:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackage-vod:List*",
        "mediapackage-vod:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediatailor:List*",
        "mediatailor:Describe*",
        "mediatailor:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "resource-groups:ListGroup",
        "resource-groups:GetGroup",
        "resource-groups:GetTags",
        "resource-groups:GetGroupQuery",
        "resource-groups:GetGroupConfiguration",
        "resource-groups:CreateGroup",
        "resource-groups:UngroupResources",
        "resource-groups:GroupResources",
        "resource-groups>DeleteGroup",
        "resource-groups:UpdateGroupQuery",
        "resource-groups:UpdateGroup",
        "resource-groups:Tag",
        "resource-groups:Untag"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "s3:"
```



```

    ],
    "Resource": "arn:aws:s3:::workflow-monitor-templates*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "sns:TagResource",
      "sns:UntagResource"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "tag:Get*",
      "tag:Describe*",
      "tag:TagResources",
      "tag:UntagResources"
    ],
    "Resource": "*"
  }
]
}

```

Política do IAM de operador

O exemplo de política a seguir destina-se a uma política do IAM do monitor de fluxo de trabalho em nível de operador. Esse perfil possibilita o acesso limitado e somente leitura aos recursos do monitor de fluxo de trabalho e aos recursos de serviço compatíveis que interagem com o monitor de fluxo de trabalho.

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:List*",
        "cloudwatch:Describe*",

```

```
        "cloudwatch:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "cloudformation:List*",
        "cloudformation:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "cloudfront:List*",
        "cloudfront:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "events:List*",
        "events:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "logs:Describe*",
        "logs:Get*"
    ],
    "Resource": "*"
},
{
```

```
    "Effect": "Allow",
    "Action": [
        "mediaconnect:List*",
        "mediaconnect:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "medialive:List*",
        "medialive:Get*",
        "medialive:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackage:List*",
        "mediapackage:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackagev2:List*",
        "mediapackagev2:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediapackage-vod:List*",
        "mediapackage-vod:Describe*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "mediatailor:List*",
```

```
        "mediatailor:Describe*",
        "mediatailor:Get*"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "s3:Get*",
        "s3:List*"
    ],
    "Resource": "arn:aws:s3:::workflow-monitor-templates*"
},
{
    "Effect": "Allow",
    "Action": [
        "tag:Get*",
        "tag:Describe*"
    ],
    "Resource": "*"
}
]
```

Grupos e modelos do monitor de fluxo de trabalho

Antes de implantar o monitoramento do fluxo de trabalho em um mapa de sinais, você deve criar os grupos e modelos para CloudWatch alarmes e EventBridge notificações. Os CloudWatch modelos definem quais cenários e limites serão usados para acionar os alarmes. Os EventBridge modelos determinarão como esses alarmes são reportados a você.

Se você quiser apenas mapear seus recursos conectados e não quiser usar os recursos do modelo de monitoramento do monitor de fluxo de trabalho, os mapas de sinais podem ser usados sem os modelos CloudWatch e EventBridge. Para ter mais informações sobre o uso de mapas de sinais, consulte [Mapas de sinais](#).

Tópicos

- [CloudWatch grupos de alarmes e modelos para monitorar seu AWS fluxo de trabalho de mídia](#)
- [EventBridge grupos de regras e modelos para monitorar seu AWS fluxo de trabalho de mídia](#)

CloudWatch grupos de alarmes e modelos para monitorar seu AWS fluxo de trabalho de mídia

Os alarmes do monitor de fluxo de trabalho permitem que você use CloudWatch as métricas existentes como base dos alarmes para seus mapas de sinal. É possível criar um grupo de modelos de alarme para classificar os tipos de alarme que são importantes para o fluxo de trabalho. Dentro de cada grupo de modelos de alarme, você cria modelos de alarme com CloudWatch métricas e parâmetros específicos que você deseja monitorar. Você pode criar seus próprios modelos de alarme ou importar modelos de alarme recomendados criados por AWS. Após a criação de um grupo de alarmes e modelos de alarme dentro desse grupo, é possível anexar um ou mais desses grupos de modelos de alarme a um mapa de sinais.

Primeiro, é necessário criar um grupo de modelos de alarme. Depois de criar um grupo de modelos de alarme, você pode criar seus próprios modelos ou usar modelos recomendados criados por AWS. Se você quiser criar seus próprios modelos de alarme, continue nesta página. Para ter mais informações sobre como importar modelos recomendados, consulte [Modelos recomendados](#).

Esta seção aborda a criação de CloudWatch alarmes usando o monitor de fluxo de trabalho. Para obter mais informações sobre como o CloudWatch serviço lida com alarmes e detalhes dos componentes do alarme, consulte: [Usando CloudWatch alarmes no Guia](#) do usuário da Amazon CloudWatch

Criar grupos de modelos de alarme

Os grupos de modelos de alarme possibilitam classificar os tipos de alarme que são importantes para o fluxo de trabalho.

Como criar um grupo de modelos de alarme

1. No painel de navegação do console do monitor de fluxo de trabalho, selecione modelos de CloudWatch alarme.
2. Selecione Criar grupo de modelos de alarme.
3. Atribua um nome exclusivo ao grupo de modelos de alarme em Nome do grupo e uma descrição opcional.
4. Selecione Criar. A página de detalhes do grupo de modelos de alarme recém-criado será aberta.

Criar modelos de alarme

Você pode criar modelos de alarme com as CloudWatch métricas e os parâmetros que deseja monitorar.

Como criar um modelo de alarme

1. Na página de detalhes do grupo de modelos de alarme, selecione Criar modelo de alarme.
2. Atribua um nome exclusivo ao modelo de alarme em Nome do modelo e uma descrição opcional.
3. Na seção Escolher métrica:
 1. Selecione um Tipo de recurso de destino. O tipo de recurso de destino é um recurso para o respectivo serviço, como um canal para MediaLive e MediaPackage ou um fluxo para MediaConnect.
 2. Selecione um nome em Nome da métrica. Essa é a CloudWatch métrica que atua como base para o alarme. A lista de métricas mudará dependendo do Tipo de recurso de destino selecionado.
4. Na seção Configurações de alarme:

Note

Para obter mais informações sobre como o CloudWatch serviço lida com alarmes e detalhes dos componentes do alarme, consulte: [Usando CloudWatch alarmes no Guia do usuário da Amazon CloudWatch](#)

1. Selecione a Estatística. Esse é o valor que será usado para monitorar a métrica; por exemplo, Soma ou Média.
2. Selecione o Operador de comparação. Esse campo faz referência ao Limite que será definido na próxima etapa.
3. Defina um Limite. Esse é um valor numérico que o Operador de comparação usa para determinar o status maior que, menor que ou igual a.
4. Defina um Período. Esse é um valor de tempo, em segundos. O Período é o intervalo em que a Estatística, o Operador de comparação e o Limite interagem para determinar se o alarme será acionado.
5. Defina os Pontos de dados. Esse valor determina quantos pontos de dados são necessários para que o alarme seja acionado.
6. Selecione como Tratar dados ausentes. Essa seleção determina como esse alarme reage a dados ausentes.

5. Selecione Criar para concluir o processo.

Um exemplo de um modelo de alarme completo pode ter os seguintes parâmetros: Um tipo de recurso alvo de MediaConnect fluxo é monitorado para o nome da métrica de desconexões. o valor Estatística está definido como Soma com um Operador de comparação “maior ou igual a” e um Limite de 10; o Período está definido como 60 segundos e requer apenas 1 em 1 Ponto de dados; e a opção Tratar dados ausentes está configurada como “ignorar”.

O resultado dessas configurações é: o monitor de fluxo de trabalho vai monitorar as desconexões no fluxo. Se dez ou mais desconexões ocorrerem em sessenta segundos, o alarme será acionado. Dez ou mais desconexões em sessenta segundos só precisam acontecer uma vez para que o alarme seja acionado.

Modelos de alarme recomendados para monitorar seu AWS fluxo de trabalho de mídia

Os modelos recomendados do monitor de fluxo de trabalho são uma seleção selecionada de métricas de serviço AWS Elemental com configurações de alarme predefinidas apropriadas para a métrica. Se você não quiser criar modelos de alarme personalizados, os modelos recomendados fornecem modelos de monitoramento de melhores práticas criados por AWS.

O monitor de fluxo de trabalho contém grupos de modelos recomendados para cada serviço compatível. Esses grupos são projetados para aplicar o monitoramento de práticas recomendadas a tipos específicos de fluxo de trabalho. Cada grupo de modelos contém alarmes cuidadosamente selecionados que são configurados com base em métricas específicas do serviço. Por exemplo, um grupo de modelos recomendado para um fluxo de trabalho MediaLive multiplex terá um conjunto diferente de métricas pré-configuradas do que um fluxo de trabalho MediaConnect CDI.

Como usar modelos de alarme recomendados

1. Siga as etapas para [criar um grupo de modelos de alarme](#) ou selecione um existente.
2. Na seção Modelos de alarme, selecione Importar. Você precisará importar os modelos recomendados da AWS para o seu grupo de modelos.
3. Use o menu suspenso de grupos de modelos de CloudWatch alarmes para selecionar um grupo AWS recomendado. Esses grupos contêm alarmes cuidadosamente selecionados para serviços específicos.
4. Selecione os modelos a serem importados usando as caixas de seleção. Cada modelo listará suas próprias métricas, com valores de monitoramento pré-configurados, e fornecerá uma descrição da métrica. Quando terminar de escolher os modelos, selecione o botão Adicionar.

5. Os modelos selecionados serão movidos para a seção Modelo(s) de alarme a serem importados. Analise suas opções e selecione Importar.
6. Depois que a importação for concluída, os modelos selecionados serão adicionados ao grupo de modelos. Para adicionar mais modelos, repita o processo de importação.
7. Os modelos importados podem ser personalizados após a importação. As configurações de alarme podem ser modificadas para atender às suas necessidades de alarme.

EventBridge grupos de regras e modelos para monitorar seu AWS fluxo de trabalho de mídia

CloudWatch usa EventBridge as regras da Amazon para enviar notificações. Primeiro, você cria um grupo de modelos de eventos. Nesse grupo de modelos de eventos, você cria modelos de eventos que determinam quais condições criam uma notificação e quem é notificado.

Esta seção aborda a criação de EventBridge regras usando o monitor de fluxo de trabalho. Para obter mais informações sobre como o EventBridge serviço usa regras, consulte: [EventBridge regras](#) no Guia do EventBridge usuário da Amazon

Criar grupos de modelos de eventos

Os grupos de modelos de eventos possibilitam classificar eventos com base no caso de uso.

Como criar um grupo de modelos de eventos

1. No painel de navegação do console do monitor de fluxo de trabalho, selecione modelos de EventBridge regras.
2. Selecione Criar grupo de modelos de eventos.
3. Atribua um nome exclusivo ao grupo de modelos de alarme em Nome do grupo e uma descrição opcional.
4. Selecione Criar. A página de detalhes do grupo de modelos de alarme recém-criado será aberta.

Criar modelos de eventos

É possível enviar notificações com base nos modelos de eventos criados.

Como criar um modelo de eventos

1. Na página de detalhes do grupo de modelos de eventos, selecione Criar modelo de evento.

2. Atribua um nome exclusivo ao modelo de evento em Nome do modelo e uma descrição opcional.
3. Na seção Configurações da regra:
 1. Selecione um Tipo de evento. Ao selecionar um tipo de evento, você pode escolher entre vários eventos criados por AWS ou selecionar Alarme ativo do mapa de sinais para usar um alarme criado por um modelo de alarme.
 2. Selecione um Serviço de destino. Isso determina como você gostaria de receber notificação sobre esse evento. Você pode selecionar Amazon Simple Notification Service ou CloudWatch registros.
 3. Depois de selecionar um serviço de destino, selecione um Destino. Esse será um tópico do Amazon SNS ou um grupo de CloudWatch registros, dependendo da seleção do serviço de destino.
4. Selecione Criar para concluir o processo.

Mapas de sinais do monitor de fluxo de trabalho

Os mapas de sinais são mapeamentos visuais de AWS recursos em seu fluxo de trabalho de mídia. É possível usar o monitor de fluxo de trabalho para iniciar a descoberta do mapa de sinais em qualquer um dos tipos de recurso compatíveis. Durante o processo de descoberta, o monitor de fluxo de trabalho mapeará automaticamente e recursivamente todos os AWS recursos conectados. Depois que o mapa de sinais for criado, você poderá usar o console do monitor de fluxo de trabalho para, por exemplo, implantar modelos de monitoramento e visualizar métricas e detalhes dos recursos associados.

Tópicos

- [Criação de mapas de sinal para AWS fluxos de trabalho de mídia](#)
- [Visualizando mapas de sinais de AWS fluxos de trabalho de mídia](#)
- [Anexando modelos de alarmes e eventos ao mapa de sinais do seu AWS fluxo de trabalho de mídia](#)
- [Implantando modelos no mapa de sinal do seu AWS fluxo de trabalho de mídia](#)
- [Atualizando o mapa de sinal do seu AWS fluxo de trabalho de mídia](#)
- [Excluindo o mapa de sinal do seu AWS fluxo de trabalho de mídia](#)

Criação de mapas de sinal para AWS fluxos de trabalho de mídia

Você pode usar mapas de sinal do monitor de fluxo de trabalho para criar um mapeamento visual de todos os AWS recursos conectados em seu fluxo de trabalho de mídia.

Como criar um mapa de sinais

1. No painel de navegação do console do monitor de fluxo de trabalho, selecione Mapas de sinais.
2. Selecione Criar mapa de sinais.
3. Atribua ao mapa de sinais um Nome e uma Descrição.
4. Na seção Descubra um novo mapa de sinais, os recursos na conta atual e a região selecionada são exibidos. Selecione um recurso para iniciar a descoberta do mapa de sinais. O recurso selecionado será o ponto de partida para a descoberta.
5. Escolha Criar. Aguarde alguns minutos para que o processo de descoberta seja concluído. Depois que o processo estiver concluído, você verá o novo mapa de sinais.

Note

As visualizações geradas no mapa de sinal do monitor de fluxo de trabalho para AWS Elemental MediaPackage canais são fornecidas pelo MediaPackage Origin Endpoint e incorrerão em taxas de transferência de dados para fora. Para saber os preços, consulte: [MediaPackagepreços](#).

Visualizando mapas de sinais de AWS fluxos de trabalho de mídia

Os mapas de sinal do monitor de fluxo de trabalho permitem que você veja um mapeamento visual de todos os AWS recursos conectados em seu fluxo de trabalho de mídia.

Visualizações do mapa de sinais

Após a seleção de um mapa de sinais, duas visualizações podem ser usadas para monitorar ou configurar o mapa de sinais. Monitorar mapa de sinal e Configurar mapa de sinal são botões sensíveis ao contexto encontrados no canto superior direito da seção do console do mapa de sinais.

Se você selecionar o mapa de sinais usando a seção Mapas de sinais do painel de navegação, ele será exibido na visualização de configuração. A visualização de configuração possibilita fazer alterações nos grupos de modelos anexados a esse mapa de sinais, implantar os modelos anexados e visualizar os detalhes básicos e as tags do mapa de sinais.

Se você selecionar o mapa de sinais usando a seção Visão geral do painel de navegação, ele será exibido na visualização de monitoramento. A visualização de monitoramento exibe os CloudWatch alarmes, EventBridge regras, alertas, registros e métricas desse mapa de sinais.

A visualização pode ser alterada a qualquer momento selecionando o botão do mapa de Monitor/Configure sinal no canto superior direito. A visualização de configuração exige permissões do IAM em nível de administrador. As permissões necessárias do IAM podem ser visualizadas aqui: [Políticas do IAM do monitor de fluxo de trabalho](#).

Percorrer o mapa de sinais

Um mapa de sinais conterá nós para cada recurso compatível da AWS descoberto pelo monitor de fluxo de trabalho. Alguns recursos, como MediaLive canais e MediaPackage endpoints, podem exibir visualizações em miniatura do conteúdo, se as visualizações em miniatura estiverem disponíveis.

Ao selecionar um nó de recurso e Visualizar detalhes do recurso selecionado no menu suspenso Ações, a página de detalhes do serviço associado será exibida. Por exemplo, selecionar um MediaLive canal e selecionar Exibir detalhes do recurso selecionado abrirá a página de detalhes do MediaLive console desse canal.

Ao selecionar um nó de recurso, será mostrada uma lista de alarmes ativos filtrada somente para esse nó. Se você selecionar o ARN de destino do recurso no alarme ativo, será exibida a página de detalhes do serviço associado, com o recurso selecionado aberto.

Anexando modelos de alarmes e eventos ao mapa de sinais do seu AWS fluxo de trabalho de mídia

Depois que você criar modelos de alarme e de evento, precisará anexá-los a um mapa de sinais. Qualquer um dos modelos de alarme e de evento que você criou pode ser anexado a qualquer mapa de sinais descoberto.

Como anexar modelos de alarme e de evento ao seu mapa de sinais

1. No painel de navegação do console do monitor de fluxo de trabalho, selecione Mapas de sinais e escolha o mapa de sinais com o qual você quer trabalhar.
2. No canto superior direito da página do mapa de sinais, na guia Grupos de modelos de CloudWatch alarmes, selecione Anexar grupos de modelos de CloudWatch alarmes.
 1. Na nova seção aberta, selecione todos os grupos de modelos de alarme que você quer aplicar a esse mapa de sinais e escolha Adicionar. Isso fará com que os grupos de modelos de alarme selecionados sejam movidos para a seção Grupos de modelos de CloudWatch alarmes anexados.

2. Ao selecionar Salvar, suas alterações serão salvas e a página do mapa de sinais será exibida.
3. À direita da página do mapa de sinais, selecione a guia Grupos de modelos de EventBridge regras e selecione Anexar grupos de modelos de EventBridge regras.
 1. Na nova seção aberta, selecione todos os grupos de modelos de evento que você deseja aplicar a esse mapa de sinais e escolha Adicionar. Isso fará com que os grupos de modelos de regras selecionados sejam movidos para a seção Grupos de modelos de EventBridge regras anexados.
 2. Ao selecionar Salvar, suas alterações serão salvas e a página do mapa de sinais será exibida.
4. Você atribuiu modelos de CloudWatch alarme e EventBridge regra ao mapa de sinais, mas o monitoramento ainda não foi implantado. A próxima seção abordará a implantação dos recursos de monitoramento.

Implantando modelos no mapa de sinal do seu AWS fluxo de trabalho de mídia

Depois que você anexar os modelos de alarme e de evento ao mapa de sinais, precisará implantar o monitoramento. O monitoramento do mapa de sinais não ficará ativo enquanto a implantação não for concluída.

O monitor de fluxo de trabalho só implantará alarmes relevantes no mapa de sinais selecionado. Por exemplo, o grupo de modelos de alarme anexado pode conter alarmes para vários serviços MediaLive, como MediaPackage, e. MediaConnect Se o mapa de sinal selecionado contiver apenas MediaLive recursos, nenhum alarme MediaPackage ou MediaConnect alarmes serão acionados.

Como implantar os modelos de monitoramento

1. Depois de anexar grupos de modelos de alarme e de evento ao mapa de sinais e salvar as alterações, selecione Implantar monitor no menu suspenso Ações.
2. Você deverá confirmar a implantação e receberá o número CloudWatch e os EventBridge recursos que serão criados. Se quiser continuar, selecione Implantar.

Note

Não há custo direto para usar o monitor de fluxo de trabalho. No entanto, há custos associados aos recursos criados e usados para monitorar o fluxo de trabalho.

Quando o monitoramento é implantado, a Amazon CloudWatch e EventBridge os recursos da Amazon são criados. Ao usar o AWS Management Console, antes de implantar o monitoramento em um mapa de sinais, você será notificado sobre quantos recursos serão criados. Para obter mais informações sobre preços, consulte: [CloudWatchpreços](#) e [EventBridge preços](#).

O monitor de fluxo de trabalho usa AWS CloudFormation modelos para implantar os EventBridge recursos CloudWatch e. Esses modelos são armazenados em um bucket de classe padrão do Amazon Simple Storage Service, que é criado em seu nome pelo monitor de fluxo de trabalho durante o processo de implantação e gera cobranças de armazenamento e recall de objetos. Para ter mais informações sobre a definição de preços, consulte [Definição de preço do Amazon S3](#).

3. O status da implantação é exibido ao lado do nome do mapa do sinais. O status de implantação também está visível na seção Pilhas do CloudFormation console. Alguns momentos após a criação e a implantação de recursos, o monitoramento do mapa de sinais será iniciado.

Atualizando o mapa de sinal do seu AWS fluxo de trabalho de mídia

Se for feita uma alteração no fluxo de trabalho, talvez seja necessário redescobrir o mapa de sinais e reimplantar os recursos de monitoramento. O monitor de fluxo de trabalho é uma ferramenta de visualização e de monitoramento que não possibilita fazer alterações no fluxo de trabalho. Os mapas de sinais representam uma visualização do fluxo de trabalho em um ponto no tempo. Caso você adicione, remova ou modifique significativamente partes do fluxo de trabalho de mídia, recomendamos redescobrir o mapa de sinais. Se você tiver recursos de monitoramento anexados ao mapa de sinais, recomendamos reimplantar o monitoramento após o processo de redescoberta.

Como redescobrir um mapa de sinais

1. No painel de navegação do console do monitor de fluxo de trabalho, selecione Mapas de sinais e escolha o mapa de sinais com o qual você quer trabalhar.
2. Verifique se você está na visualização Configurar mapa de sinal. Para ter mais informações sobre como alterar visualizações, consulte [Visualizar mapas de sinais](#).
3. No canto superior direito da página do mapa de sinais, selecione o menu suspenso Ações. Selecione Redescobrir.
4. Você verá a tela de redescoberta. Selecione um recurso integrante do fluxo de trabalho que você está redescobrando. Selecione o botão Redescobrir.

5. O mapa de sinais será reconstruído de acordo com o fluxo de trabalho atual. Se você precisar reimplantar os recursos de monitoramento, permaneça na página desse mapa de sinais. Todos os modelos de monitoramento anexados anteriormente permanecerão anexados, mas precisarão ser reimplantados.

Como reimplantar modelos de monitoramento após a redescoberta de um mapa de sinais

1. Após a redescoberta, o mapa de sinais atualizado será exibido. Para reimplantar os modelos de monitoramento, selecione Implantar monitor no menu suspenso Ações.
2. Você deverá confirmar a implantação e receberá o número de todos CloudWatch os EventBridge recursos que serão criados. Se quiser continuar, selecione Implantar.
3. O status da implantação é exibido ao lado do nome do mapa do sinais. Alguns momentos após a criação e a implantação de recursos, o monitoramento do mapa de sinais será iniciado.

Excluindo o mapa de sinal do seu AWS fluxo de trabalho de mídia

Se você não precisar mais de um mapa de sinais, ele poderá ser excluído. Se você tiver modelos de monitoramento implantados no mapa de sinal, o processo de exclusão solicitará que você exclua todos CloudWatch os EventBridge recursos que foram implantados nesse mapa de sinal. A exclusão dos recursos implantados não afeta os modelos que os criaram. Essa exclusão de recursos é para garantir que você não tenha CloudWatch EventBridge recursos implantados, mas não usados.

Como excluir um mapa de sinais

1. No painel de navegação do console do monitor de fluxo de trabalho, selecione Mapas de sinais e o botão de opção ao lado do mapa de sinais a ser excluído.
2. Selecione o botão Excluir. Você precisará confirmar a exclusão dos recursos de monitoramento. Selecione Excluir para iniciar o processo de exclusão de recursos de monitoramento.
3. A coluna Monitorar implantação exibirá o status atual. Quando o status for alterado para DELETE_COMPLETE, selecione novamente o botão Excluir.
4. Será necessário confirmar a exclusão do mapa de sinais. Selecione Excluir para excluir o mapa de sinais.

Cotas do monitor de fluxo de trabalho

A seção a seguir contém cotas para recursos de monitoramento de fluxo de trabalho. Cada cota é definida “por conta”. Se precisar aumentar a cota da sua conta, você pode usar o console [AWS Service Quotas](#) para solicitar um aumento, a menos que indicado de outra forma na tabela a seguir.

Cotas

Tipo de atributo	Quota
CloudWatch grupos de modelos de alarme	20
CloudWatch modelos de alarme	200
EventBridge grupos de modelos de regras	20
EventBridge modelos de regras	200
Mapas de sinais	30
Mapas de sinal: grupos de modelos de CloudWatch alarme anexados a um único mapa de sinal	5 Você não pode aumentar essa cota.
Mapas de sinais: grupos de modelos de EventBridge regras anexados a um único mapa de sinal	5 Você não pode aumentar essa cota.

Usar o monitor de fluxo de trabalho

Use as seções Visão geral e Mapas de sinais do console do monitor de fluxo de trabalho para analisar o status atual dos fluxos de trabalho e quaisquer alarmes, métricas e logs associados.

Tópicos

- [Visão geral do monitor de fluxo de trabalho](#)
- [Visão geral dos logs e das métricas do monitor de fluxo de trabalho](#)
- [Usar mapas de sinais do monitor de fluxo de trabalho](#)

Visão geral do monitor de fluxo de trabalho

A seção Visão geral do console do monitor de fluxo de trabalho é um painel que fornece informações resumidas sobre os mapas de sinais. Na seção de visão geral, você pode ver o estado atual do monitoramento de cada mapa de sinal, bem como as CloudWatch métricas e quaisquer CloudWatch registros associados. Ao selecionar qualquer mapa de sinais, a página do console correspondente será exibida.

Filtragem de visão geral

Usando a barra de pesquisa na seção de visão geral, é possível filtrar a lista de mapas de sinais usando restrições sensíveis ao contexto. Depois de selecionar a barra de pesquisa, você verá uma lista de Propriedades para usar como filtro. Ao selecionar uma propriedade, serão apresentados Operadores, como Igual, Contém, Não é igual e Não contém. Ao selecionar um operador, será criada uma lista de recursos do tipo de propriedade selecionado. A seleção de um desses recursos fará com que a lista de mapas de sinais exiba somente mapas que se encaixem na restrição definida.

Visão geral dos logs e das métricas do monitor de fluxo de trabalho

Para visualizar CloudWatch métricas e registros de um mapa de sinal, selecione o botão de rádio ao lado do nome do mapa de sinal. Uma interface com guias para métricas e logs será exibida abaixo da lista de mapas de sinais.

CloudWatch Métricas

CloudWatch as métricas para o mapa de sinal selecionado serão sensíveis ao contexto e exibirão apenas métricas associadas aos serviços usados nesse fluxo de trabalho de mapas de sinais. É possível usar as ferramentas de métricas na tela para personalizar os períodos de métrica e os intervalos de tempo exibidos.

CloudWatch Registros

Se você associou um grupo de CloudWatch registros ao mapa de sinal, esse grupo será exibido aqui.

Usar mapas de sinais do monitor de fluxo de trabalho

Na seção Visão geral do console, é possível selecionar um mapa de sinais específico para ver mais informações sobre ele e os respectivos recursos de monitoramento anexados.

Ao selecionar um mapa, ele é exibido, apresentando várias seções com guias que contêm mais informações:

- CloudWatch alarmes
- EventBridge regras
- AWS Alertas elementares
- Metrics
- Logs
- Detalhes básicos

Percorrer o mapa de sinais

Um mapa de sinais conterá nós para cada recurso compatível da AWS descoberto pelo monitor de fluxo de trabalho. Alguns recursos, como MediaLive canais e MediaPackage endpoints, podem exibir visualizações em miniatura do conteúdo, se as visualizações em miniatura estiverem disponíveis.

Ao selecionar um nó de recurso e Visualizar detalhes do recurso selecionado no menu suspenso Ações, a página de detalhes do serviço associado será exibida. Por exemplo, selecionar um MediaLive canal e selecionar Exibir detalhes do recurso selecionado abrirá a página de detalhes do MediaLive console desse canal.

Ao selecionar um nó de recurso, será mostrada uma lista de alarmes ativos filtrada somente para esse nó. Se você selecionar o ARN de destino do recurso no alarme ativo, será exibida a página de detalhes do serviço associado, com o recurso selecionado aberto.

Dual-stack Endpoints (IPv6) para AWS Elemental MediaTailor

AWS Elemental MediaTailor oferece suporte a endpoints de pilha dupla que você pode usar para se conectar via IPv4 ou IPv6. Esses endpoints são funcionalmente equivalentes aos endpoints padrão IPv4-only. O comportamento do serviço é idêntico e somente a camada de transporte de rede é diferente.

O que são endpoints de pilha dupla?

Dual-stack os endpoints anunciam endereços IPv4 e IPv6. Quando você usa um endpoint de pilha dupla, o DNS é resolvido para um endereço IPv4 ou IPv6 com base na configuração da rede e nas preferências do cliente. Você não precisa alterar seus MediaTailor recursos, configurações de reprodução ou comportamento de inserção de anúncios.

Recursos compatíveis

A tabela a seguir lista o suporte de MediaTailor recursos para endpoints de pilha dupla.

Recurso	Dual-stack apoio
APIs do plano de controle (todas as regiões)	Compatível
Plano de dados SSAI (inicialização da sessão, manifesto, segmento)	Compatível
Plano de dados de montagem de canais	Compatível
Console de gerenciamento da AWS	Compatível
AWS SDK	Em breve

Dual-stack formato de endpoint

MediaTailor endpoints de pilha dupla usam o domínio: `.api.aws`

- Plano de controle: `mediatailor.region.api.aws`
- Plano de dados SSAI: exibido na seção de endpoints Dualstack dos detalhes da configuração de reprodução no Console de gerenciamento da AWS
- Plano de dados de montagem do canal: exibido nos detalhes de saída do seu canal no Console de gerenciamento da AWS.

Seus aplicativos existentes continuam trabalhando com os IPv4-only endpoints originais (`api.mediatailor.region.amazonaws.com` para plano de controle, `unique-id.mediatailor.region.amazonaws.com` para plano de dados). Para obter uma lista completa dos endpoints disponíveis, consulte [MediaTailor endpoints e cotas na AWS Referência](#) geral.

Usar endpoints de pilha dupla

Console de gerenciamento da AWS

Dual-stack os endpoints aparecem na página de detalhes da configuração de reprodução, na seção de endpoints do Dualstack. Você pode copiar esses URLs diretamente para uso em sua configuração de player ou CDN.

AWS Command Line Interface

Especifique o URL do endpoint de pilha dupla ao fazer chamadas de API:

```
aws mediatailor get-playback-configuration \  
  --name my-config \  
  --endpoint-url https://mediatailor.us-east-1.api.aws
```

my-config Substitua pelo nome da sua configuração de reprodução e *us-east-1* pela sua AWS região.

AWS SDK

Ative o dual-stack na configuração do SDK.

Python (boto3):

```
import boto3
```

```
client = boto3.client(  
    'mediatailor',  
    endpoint_url='https://mediatailor.us-east-1.api.aws'  
)
```

JavaScript (AWS SDK v3):

```
import { MediaTailorClient } from "@aws-sdk/client-mediatailor";  
  
const client = new MediaTailorClient({  
    useDualstackEndpoint: true  
});
```

Considerações

Lembre-se do seguinte ao usar endpoints de pilha dupla:

- Dual-stack os endpoints estão disponíveis em todos os Regiões da AWS lugares MediaTailor disponíveis.
- Você não precisa alterar suas configurações de reprodução, do servidor de decisão de anúncios ou da CDN para usar endpoints de pilha dupla.
- Se sua CDN usa roteamento baseado em padrões de nome de host, verifique se suas regras acomodam o formato do domínio. `.api.aws`
- A AWS orientação padrão para terminais de pilha dupla se aplica. Para obter mais informações, consulte [Dual-stack os endpoints FIPS](#) na Referência AWS geral.

Cotas em AWS Elemental MediaTailor

MediaTailor as solicitações de recursos e operações estão sujeitas às seguintes cotas (anteriormente chamadas de “limites”).

Você pode usar o AWS serviço Service Quotas para visualizar cotas e solicitar aumentos de cotas MediaTailor, além de muitos outros serviços. AWS Para acessar mais informações, consulte o [Guia do usuário do Service Quotas](#).

Cotas na inserção de anúncios

A tabela a seguir descreve as cotas na inserção de AWS Elemental MediaTailor anúncios. Salvo indicação em contrário, as cotas não são ajustáveis.

Nome	Valor de cota padrão	Description	
Tamanho do Ad Decision Server (ADS – Servidor de decisões de anúncios)	25.000	O número máximo de caracteres na especificação de um ADS.	
Redirecionamentos do ADS	7	A profundidade máxima de redirecionamentos que MediaTailor segue nas tags de wrapper VAST. MediaTailor desiste se houver redirecionamentos adicionais.	
Tempo limite do ADS	3	O número máximo de segundos de MediaTailor espera antes de atingir o tempo limite em uma conexão aberta	

Nome	Valor de cota padrão	Description	
		com um servidor de decisão de anúncios (ADS). Quando uma conexão expira devido à falta de resposta do MediaTailor ADS, não consegue preencher o anúncio disponível com anúncios.	
Solicitações de inserção de anúncios	10.000	O máximo de solicitações por segundo para fazer manifestos personalizados ao realizar a inserção de anúncios no servidor. A inserção de anúncios processa solicitações recebidas de manifestos, inicialização de sessão, dados de rastreamento e segmentos de anúncios. Essa cota é ajustável.	
Configurações	1.000	O número máximo de configurações que MediaTailor permite.	

Nome	Valor de cota padrão	Description	
Tamanho da origem do conteúdo	512	O número máximo de caracteres em uma especificação de origem de conteúdo.	
Tempo limite do servidor de origem do conteúdo	2	O número máximo de segundos que MediaTailor espera antes de atingir o tempo limite em uma conexão aberta com o servidor de origem do conteúdo ao solicitar manifestos de modelo. Os tempos limite geram erros de resposta HTTP 504 (GatewayTimeoutException) .	
Tamanho do manifesto	2	O tamanho máximo, em MB, de qualquer manifesto de reprodução de origem. Para garantir que você permaneça abaixo da cota, use gzip para compactar os manifestos de entrada no MediaTailor.	

Nome	Valor de cota padrão	Description	
Configurações do pacote	5	O número máximo de configurações de pacotes por fonte (seja ao vivo ou vídeo sob demanda).	
Cronogramas de pré-busca	25	O número máximo de agendamentos de pré-busca ativos por configuração de reprodução. Os agendamentos de pré-busca expirados não contam para esse limite.	

Nome	Valor de cota padrão	Description	
Tempo limite de solicitação de beacon de relatórios do lado do servidor	3 segundos	O número máximo de segundos que MediaTailor espera antes de atingir o tempo limite em uma conexão aberta com o servidor ao disparar um beacon para emissão de relatórios do lado do servidor. Quando uma conexão expira, MediaTailor não consegue disparar o beacon e o serviço registra uma <code>ERROR_FIRING_BEACON_FAILED</code> mensagem no MediaTailor/AdDecisionServerInteractionslogin. CloudWatch	

Nome	Valor de cota padrão	Description	
Tempo limite do gancho PRE_SESSION_INITIALIZATION	2 segundos	O número máximo de segundos que a execução de um hook PRE_SESSION_INITIALIZATION pode levar, incluindo todas as etapas da função e chamadas HTTP. Se o gancho exceder seu tempo limite, MediaTailor descarta toda a saída e prossegue como se nenhuma função tivesse sido executada.	
Tempo limite do gancho PRE_AD_REQUEST	2 segundos	O número máximo de segundos que uma execução de gancho PRE_AD_REQUEST pode levar, incluindo todas as etapas da função e chamadas HTTP. Se o gancho exceder seu tempo limite, MediaTailor descarta toda a saída e prossegue como se nenhuma função tivesse sido executada.	

Nome	Valor de cota padrão	Description	
Funções máximas por gancho	20	O número máximo de execuções totais de funções permitidas por gancho.	
Expiração da sessão	10 vezes a duração do manifesto	A quantidade máxima de tempo que MediaTailor permite que uma sessão permaneça inativa antes de terminar a sessão. A atividade da sessão pode ser uma solicitação de player ou um adiantamento do servidor de origem. Quando a sessão expira, MediaTailor retorna um erro de HTTP 400 (Bad Request) resposta.	

Cotas na montagem do canal

A tabela a seguir descreve as cotas na montagem AWS Elemental MediaTailor do canal. Salvo indicação em contrário, as cotas são [ajustáveis](#).

Nome	Valor de cota padrão	Description	
Solicitações de manifesto de canais, por conta	400	O número máximo de solicitações de manifesto de saída por segundo para	

Nome	Valor de cota padrão	Description	
		todos os canais de montagem de canais em uma conta.	
Solicitações de manifesto de canais, por canal	50	O número máximo de solicitações de manifesto de saída por segundo para qualquer canal único de montagem de canais.	
Saídas de canal	5	O número máximo de saídas por canal.	
Canais por conta	100	O número máximo de canais por conta.	
Fontes ao vivo	50	O número máximo de fontes ativas para o local de origem.	
Programas por canal	400	O número máximo de programas por canal.	
Configurações de entrega por segmento	5	O número máximo de configurações de entrega de segmentos por local de origem.	
Locais de origem	50	O número máximo de locais de origem por conta.	

Nome	Valor de cota padrão	Description	
Fontes de VOD	1.000	O número máximo de fontes de vídeo sob demanda (VOD) para o local de origem.	

A tabela a seguir descreve os limites de limitação na montagem do AWS Elemental MediaTailor canal. Salvo indicação em contrário, as cotas são [ajustáveis](#).

Nome	Limite máximo padrão de transações por segundo	Description	
ConfigureLogsForChannel	1	Configure os registros para o canal.	
CreateChannel	1	Crie um canal.	
CreateLiveSource	1	Crie uma fonte ativa.	
CreateProgram	3	Crie um programa.	
CreateSourceLocation	1	Crie um local de origem.	
CreateVodSource	1	Crie uma fonte de VOD.	
DeleteChannel	1	Exclua um canal.	
DeleteChannelPolicy	1	Exclua uma política de canal.	
DeleteLiveSource	1	Exclua uma fonte ativa.	
DeleteProgram	3	Exclua um programa.	

Nome	Limite máximo padrão de transações por segundo	Description	
DeleteSourceLocation	1	Exclua um local de origem.	
DeleteVodSource	1	Exclua uma fonte de VOD.	
DescribeChannel	5	Descreva um canal.	
DescribeLiveSource	5	Descreva uma fonte ativa.	
DescribeProgram	5	Descreva um programa.	
DescribeSourceLocation	5	Descreva um local de origem.	
DescribeVodSource	5	Descreva uma fonte de VOD.	
GetChannelPolicy	5	Obtenha uma política de canal.	
GetChannelSchedule	5	Obtenha uma programação de canais.	
ListAlerts	5	Listar alertas.	
ListChannels	5	Listar canais.	
ListLiveSources	5	Liste fontes ao vivo.	
ListPrograms	5	Listar programas.	

Nome	Limite máximo padrão de transações por segundo	Description	
ListSourceLocations	5	Liste os locais de origem.	
ListTagsForResource	5	Listar as etiquetas de um recurso.	
ListVodSources	5	Listar fontes de VOD.	
PutChannelPolicy	3	Coloque uma política de canal.	
StartChannel	1	Inicie um canal.	
StopChannel	1	Pare um canal.	
TagResource	1	Marca um recurso.	
UntagResource	1	Desmarca um recurso.	
UpdateChannel	1	Atualize um canal.	
UpdateLiveSource	1	Atualize uma fonte ativa.	
UpdateProgram	1	Atualize um programa.	
UpdateSourceLocation	1	Atualize um local de origem.	
UpdateVodSource	1	Atualize uma fonte de VOD.	

Solução de problemas MediaTailor

Se você encontrar erros de reprodução ou dificuldades semelhantes ao trabalhar com AWS Elemental MediaTailor, consulte os tópicos desta seção.

Tópicos

- [Solução de problemas de fluxo de MediaTailor eventos](#)
- [Solução de problemas de reprodução de MediaTailor](#)
- [Desduplicação de beacons](#)
- [Solução de problemas de parâmetros](#)
- [MediaTailor guia de solução de problemas para ignorar anúncios](#)

Solução de problemas de fluxo de MediaTailor eventos

Compreender o fluxo de AWS Elemental MediaTailor eventos fornece uma base poderosa para solucionar problemas de inserção de anúncios. Ao analisar a sequência, o tempo e os padrões dos eventos, você pode identificar rapidamente onde os problemas ocorrem e implementar soluções específicas.

Esta seção fornece orientação prática para usar a análise de fluxo de eventos para diagnosticar problemas. Para entender os conceitos básicos de fluxo de eventos, consulte [Fluxo de eventos de inserção de anúncios](#).

Identificação de fluxos de eventos incompletos

Fluxos de eventos incompletos ocorrem quando a sequência esperada de eventos é interrompida antes de alcançar a personalização bem-sucedida do manifesto (o processo de MediaTailor inserir informações personalizadas do anúncio no manifesto). Identificar onde o fluxo é interrompido ajuda a identificar a causa raiz das falhas na inserção de anúncios.

Padrões comuns de fluxo incompletos

Pontos de falha diferentes no fluxo de eventos indicam tipos específicos de problemas, como os seguintes.

- O fluxo é interrompido após a detecção da oportunidade de anúncio: indica problemas com os marcadores do anúncio ou com o próprio manifesto que MediaTailor impedem a realização de uma

solicitação de ADS. Problemas de conectividade, configuração ou tempo limite do ADS ocorreriam após a solicitação do ADS ser feita.

- O fluxo é interrompido após a solicitação do ADS: sugere problemas de resposta do ADS, problemas de análise do VAST, falhas no processamento criativo, tempos limite do ADS, erros de conectividade ou problemas de configuração, como URLs inválidos do ADS, que só são descobertos quando a solicitação é feita.
- Sinalizador de rastreamento ausente: pode indicar problemas de configuração de rastreamento, problemas de relatórios do lado do servidor ou lacunas na implementação do lado do cliente.

CloudWatch consultas para análise de fluxo incompleta

Use essas consultas do Amazon CloudWatch Logs Insights para identificar fluxos de eventos incompletos. Execute essas consultas nos grupos de registros apropriados com base no tipo de análise necessária.

Seleção do grupo de registros:

- MediaTailor/AdDecisionServerInteractions- Use para consultas que analisam interações do servidor de decisão de anúncios, oportunidades de anúncios e ADS-related falhas.
- MediaTailor/TranscodeService- Use para analisar problemas em que os anúncios não foram inseridos devido a problemas de transcodificação, falhas no processamento criativo ou outros problemas não relacionados a anúncios.

Exemplo identifique oportunidades de anúncios sem uma personalização bem-sucedida do manifesto

Grupo de registros: MediaTailor/AdDecisionServerInteractions

A consulta a seguir identifica oportunidades de anúncios que não resultaram na personalização bem-sucedida do manifesto:

```
fields @timestamp, eventType, avail.availId, sessionId
| filter eventType = "AD_MARKER_FOUND"
| stats count() as total_opportunities by avail.availId
| join (
    fields @timestamp, eventType, avail.availId
    | filter eventType = "FILLED_AVAIL"
    | stats count() as successful_fills by avail.availId
) on avail.availId
```

```
| where ispresent(total_opportunities) and not ispresent(successful_fills)
| sort total_opportunities desc
```

Exemplo analisar as taxas de conclusão do fluxo de eventos

Grupo de registros: MediaTailor/AdDecisionServerInteractions

A consulta a seguir analisa as taxas de conclusão em diferentes tipos de eventos:

```
fields @timestamp, eventType, avail.availId
| filter eventType in ["AD_MARKER_FOUND", "MAKING_ADS_REQUEST", "VAST_RESPONSE",
"FILLED_AVAIL", "BEACON_FIRED"]
| stats count() by eventType, avail.availId
| sort avail.availId, eventType
```

Exemplo encontre sessões com eventos de beacon ausentes

Grupo de registros: MediaTailor/AdDecisionServerInteractions

A consulta a seguir identifica sessões que preencheram os avais, mas nenhum evento de beacon correspondente:

```
fields @timestamp, eventType, sessionId, avail.availId
| filter eventType = "FILLED_AVAIL"
| stats count() as filled_avails by sessionId
| join (
  fields @timestamp, eventType, sessionId
  | filter eventType = "BEACON_FIRED"
  | stats count() as beacon_events by sessionId
) on sessionId
| where filled_avails > 0 and (not ispresent(beacon_events) or beacon_events = 0)
| sort filled_avails desc
```

Exemplo identificar falhas de inserção e transcodificação

Grupo de registros: MediaTailor/TranscodeService

A consulta a seguir identifica problemas de transcodificação que impedem a inserção bem-sucedida do anúncio:

```
fields @timestamp, eventType, sessionId, requestId
```

```
| filter eventType in ["TRANSCODE_IN_PROGRESS", "INTERNAL_ERROR", "MISSING_VARIANTS",  
"PROFILE_NOT_FOUND"]  
| stats count() as transcode_issues by eventType, sessionId  
| sort transcode_issues desc
```

Análise de problemas de cronometragem de eventos

A análise do tempo de eventos ajuda a identificar gargalos de desempenho e otimizar os fluxos de trabalho de inserção de anúncios. Padrões de tempo incomuns geralmente indicam problemas subjacentes que afetam a experiência do espectador.

Limites de tempo de desempenho

Use esses limites de tempo para identificar possíveis problemas de desempenho.

- Duração total do fluxo de mais de 5 segundos: pode afetar a experiência do espectador e pode indicar problemas de desempenho do ADS, problemas no servidor de origem (como tempos limite de recuperação de manifestos) ou problemas internos, incluindo MediaTailor problemas de infraestrutura com NAT Gateway, DynamoDB, EC2 ou outros componentes do sistema.
- Tempo de resposta do ADS superior a 2 segundos: sugere problemas de desempenho do ADS ou problemas de latência da rede.
- Personalização do manifesto por mais de 1 segundo: pode indicar atrasos no processamento criativo, problemas no servidor de origem (como tempos limite de recuperação de manifestos) ou problemas internos do MediaTailor sistema, incluindo restrições de infraestrutura com NAT Gateway, DynamoDB, EC2 ou outros componentes.

Consultas de análise de tempo

Use essas consultas para analisar os padrões de temporização de eventos.

Exemplo medir a duração total do fluxo de eventos

A consulta a seguir mede a duração total dos fluxos de eventos e identifica aqueles que excedem 5 segundos:

```
fields @timestamp, eventType, avail.availId  
| filter avail.availId = "your-avail-id"  
| filter eventType in ["AD_MARKER_FOUND", "FILLED_AVAIL"]  
| sort @timestamp asc
```

```
| stats min(@timestamp) as start_time, max(@timestamp) as end_time by avail.availId  
| eval duration_seconds = (end_time - start_time) / 1000  
| where duration_seconds > 5
```

Example analise o tempo de resposta do ADS

A consulta a seguir analisa os tempos de resposta do ADS e identifica aqueles que excedem 2 segundos:

```
fields @timestamp, eventType, avail.availId  
| filter avail.availId = "your-avail-id"  
| filter eventType in ["MAKING_ADS_REQUEST", "VAST_RESPONSE"]  
| sort @timestamp asc  
| stats min(@timestamp) as request_time, max(@timestamp) as response_time by  
  avail.availId  
| eval ads_response_seconds = (response_time - request_time) / 1000  
| where ads_response_seconds > 2
```

Example identifique a personalização lenta do manifesto

A consulta a seguir identifica processos de personalização de manifestos que levam mais de 1 segundo:

```
fields @timestamp, eventType, avail.availId  
| filter avail.availId = "your-avail-id"  
| filter eventType in ["VAST_RESPONSE", "FILLED_AVAIL"]  
| sort @timestamp asc  
| stats min(@timestamp) as response_time, max(@timestamp) as filled_time by  
  avail.availId  
| eval personalization_seconds = (filled_time - response_time) / 1000  
| where personalization_seconds > 1
```

Problemas e soluções comuns de fluxo de eventos

Esta seção fornece soluções para problemas de fluxo de eventos frequentemente encontrados, organizados por tipo de problema e sintomas.

Falhas na solicitação do servidor de decisão de anúncios

Sintomas: o fluxo de eventos é interrompido após a detecção da oportunidade de anúncio. Nenhum evento de solicitação de ADS registrado.

Causas e soluções comuns

- Erros de configuração do URL do ADS: verifique se o URL do ADS na sua configuração de reprodução está correto e acessível. No registro de interação com anúncios, você verá um evento de solicitação do ADS (MAKING_ADS_REQUEST), mas nenhuma resposta VAST correspondente, geralmente acompanhada por um evento de erro `ERROR_UNKNOWN` ou similar.
- Problemas de conectividade de rede: verifique a conectividade de rede entre MediaTailor e seu ADS, incluindo regras de firewall e resolução de DNS.
- SSL/TLS problemas de certificado: garanta que seu ADS use certificados SSL válidos de uma autoridade de certificação confiável. Especificamente para o Google Ad Manager, talvez seja necessário entrar em contato com o [AWS Support](#) para ativar um sinalizador de configuração que aceite os certificados SSL do Google.

Consulta de diagnóstico

A consulta a seguir ajuda a diagnosticar falhas na solicitação do ADS rastreando a sequência de eventos:

```
fields @timestamp, eventType, sessionId
| filter sessionId = "your-session-id"
| filter eventType in ["AD_MARKER_FOUND", "MAKING_ADS_REQUEST", "ERROR_ADS_IO",
"ERROR_UNKNOWN_HOST"]
| sort @timestamp asc
```

Falhas de resposta do servidor de decisão de anúncios

Sintomas: as solicitações do ADS são bem-sucedidas, mas MediaTailor não recebem uma resposta, ou ocorrem erros de análise.

Causas e soluções comuns

- Formato VAST inválido: valide suas respostas do ADS VAST em relação aos padrões de especificação VAST.
- Problemas de tempo limite do ADS: aumente as configurações de tempo limite do ADS ou otimize o tempo de resposta do ADS.
- Inventário de anúncios vazio: verifique a disponibilidade do inventário de anúncios e os critérios de segmentação na sua configuração do ADS.

Consulta de diagnóstico

A consulta a seguir ajuda a diagnosticar falhas de resposta do ADS examinando eventos de solicitação e resposta:

```
fields @timestamp, eventType, sessionId
| filter sessionId = "your-session-id"
| filter eventType in ["MAKING_ADS_REQUEST", "VAST_RESPONSE", "EMPTY_VAST_RESPONSE",
"ERROR_ADS_RESPONSE_PARSE", "ERROR_ADS_TIMEOUT"]
| sort @timestamp asc
```

Falhas manifestas de personalização

Sintomas: respostas VAST recebidas, mas a personalização manifesta falha ou os anúncios são ignorados.

Causas e soluções comuns:

- Problemas de transcodificação criativa: verifique se o anúncio é um `NEW_CREATIVE`, o que requer transcodificação antes da inserção. Você também pode verificar se há erros de transcodificação examinando o MediaTailor/TranscodeService registro em busca de eventos de erro, como `INTERNAL_ERROR` ou `MISSING_VARIANTS`, `PROFILE_NOT_FOUND`
- Problemas de incompatibilidade de duração: verifique se as durações dos anúncios se encaixam nas durações de intervalos disponíveis.
- Problemas de limite de personalização: revise as configurações de limite de personalização em sua configuração de reprodução.

Consulta de diagnóstico

A consulta a seguir ajuda a diagnosticar falhas manifestas de personalização examinando respostas VAST e disponibilidades preenchidas:

```
fields @timestamp, eventType, sessionId, skippedAds
| filter sessionId = "your-session-id"
| filter eventType in ["VAST_RESPONSE", "FILLED_AVAIL", "WARNING_NO_ADVERTISEMENTS"]
| sort @timestamp asc
```

Consulta por motivos de ignoração do anúncio

A consulta a seguir fornece informações detalhadas sobre por que os anúncios foram ignorados:

```
fields @timestamp, eventType, sessionId, skippedAds.reason, skippedAds.creativeUniqueId
| filter sessionId = "your-session-id"
| filter eventType = "WARNING_NO_ADVERTISEMENTS" or ispresent(skippedAds)
| sort @timestamp asc
```

Consulte os motivos do anúncio ignorado e IDs exclusivos criativos

A consulta a seguir fornece informações detalhadas sobre anúncios ignorados, incluindo motivos e IDs criativos exclusivos para os dois primeiros anúncios em cada disponibilidade:

```
fields @timestamp, eventType
| filter sessionId = "your-session-id"
| filter eventType = "FILLED_AVAIL"
| fields avail.skippedAds.0.vastDuration as SkippedDur_Ad0,
  avail.skippedAds.0.skippedReason as Ad0_SkipReason,
  avail.skippedAds.0.creativeUniqueId as SkippedCreative0_UID
| fields avail.skippedAds.1.vastDuration as SkippedDur_Ad1,
  avail.skippedAds.1.skippedReason as Ad1_SkipReason,
  avail.skippedAds.1.creativeUniqueId as SkippedCreative1_UID
| sort @timestamp desc
```

Rastreamento de falhas de beacon

Sintomas: Personalização bem-sucedida do manifesto, mas beacons de rastreamento ausentes ou com falha.

Causas e soluções comuns

- Client-side problemas de implementação: a maioria dos problemas de beacon de rastreamento resulta de problemas de implementação do lado do cliente, como não pesquisar URLs de rastreamento com frequência suficiente para rastreamento do lado do cliente ou problemas de lógica de acionamento de beacon específicos do jogador.
- Problemas de acessibilidade do URL de rastreamento: verifique se os URLs de rastreamento nas respostas do VAST estão acessíveis e retorne as respostas apropriadas. Podem ocorrer problemas quando os URLs não estão acessíveis ou quando há problemas internos que impedem o MediaTailor rastreamento bem-sucedido da entrega das respostas.
- Problemas na solicitação do segmento do player: falhas aparentes do farol de rastreamento podem ocorrer quando o player cliente não solicita realmente nenhum segmento. Isso faz com que

nenhum beacon seja enviado, o que parece uma falha de rastreamento, mas na verdade é um problema de implementação do jogador e não um problema de farol.

Consulta de diagnóstico

A consulta a seguir ajuda a diagnosticar falhas de beacon de rastreamento examinando disponibilidades preenchidas e eventos de beacon:

```
fields @timestamp, eventType, sessionId
| filter sessionId = "your-session-id"
| filter eventType in ["FILLED_AVAIL", "BEACON_FIRED", "ERROR_FIRING_BEACON_FAILED"]
| sort @timestamp asc
```

Melhores práticas de monitoramento de fluxo de eventos

Implemente essas práticas de monitoramento para identificar e resolver proativamente problemas de fluxo de eventos:

Configurando CloudWatch alarmes

Crie CloudWatch alarmes da Amazon para monitorar as principais métricas de fluxo de eventos.

- Alarme de taxa de conclusão de fluxo: alerta quando a proporção entre a personalização bem-sucedida do manifesto e as oportunidades de anúncios cair abaixo dos limites aceitáveis.
- Alarme de tempo de resposta do ADS: monitore os tempos médios de resposta do ADS e alerte quando eles excederem os limites de desempenho.
- Alarme de taxa de erro: rastreie as frequências de eventos de erro e alerte sobre picos incomuns em tipos de erro específicos.

Consultas regulares de monitoramento

Execute essas consultas regularmente para manter a visibilidade da integridade do fluxo de eventos:

Exemplo taxa de sucesso do fluxo diário de eventos

A consulta a seguir fornece uma visão geral diária das taxas de sucesso do fluxo de eventos por tipo de evento:


```
fields @timestamp, eventType
| filter @timestamp > datefloor(@timestamp, 1d)
| stats count() as total_events by eventType
| sort total_events desc
```

Exemplo tendência da taxa de erro por hora

A consulta a seguir rastreia as taxas de erro por hora para identificar problemas populares:

```
fields @timestamp, eventType
| filter eventType like /ERROR_/
| stats count() as error_count by datefloor(@timestamp, 1h) as hour
| sort hour desc
```

Orientação de otimização de desempenho

Use a análise de fluxo de eventos para otimizar o desempenho da inserção de anúncios.

- Otimização de ADS: trabalhe com seu provedor de ADS para otimizar os tempos de resposta e reduzir a latência.
- Preparação criativa: Pre-transcode anúncios criativos que correspondam aos seus perfis de conteúdo e reduzam os atrasos no processamento.
- Ajuste de configuração: ajuste as configurações de tempo limite, limites de personalização e outros parâmetros de configuração com base na análise do fluxo de eventos.

Recursos adicionais para solução de problemas

Para obter orientação adicional sobre solução de problemas além da análise do fluxo de eventos:

- Para obter informações detalhadas sobre o formato de registro e especificações técnicas, consulte [Visualizar logs](#).
- Para uma solução abrangente de problemas comuns de inserção de anúncios, consulte [Solução de problemas comuns do](#).
- Para obter orientações de configuração de monitoramento e alerta, consulte [Monitoramento AWS Elemental MediaTailor com CloudWatch métricas da Amazon](#).
- Para procedimentos de registro de depuração, consulte [Gerando registros de depuração](#).

Solução de problemas de reprodução de MediaTailor

Esta seção fornece informações sobre os códigos de erro HTTP que você pode receber ao testar o software do player e durante o processamento normal de solicitações do player.

Note

Você também pode receber erros da AWS Elemental MediaTailor API durante operações de configuração como `PutPlaybackConfiguration` `GetPlaybackConfiguration` e. Para obter informações sobre esses tipos de erros, consulte a [Referência AWS Elemental MediaTailor da API](#).

Quando seu player envia uma solicitação para AWS Elemental MediaTailor, diretamente ou por meio de uma CDN, MediaTailor responde com um código de status. Se a solicitação for processada MediaTailor com êxito, ela retornará o código de status HTTP 200 OK, indicando sucesso, junto com o manifesto preenchido. Se a solicitação não for bem-sucedida, MediaTailor retornará um código de status HTTP, um nome de exceção e uma mensagem de erro.

AWS Elemental MediaTailor retorna duas classes de erros:

- Erros do cliente — erros que geralmente são causados por um problema na própria solicitação, como uma solicitação formatada incorretamente, um parâmetro inválido ou uma URL incorreta. Esses erros têm um código de resposta HTTP 4xx.
- Erros de servidor — erros que geralmente são causados por um problema com MediaTailor ou uma de suas dependências, como o servidor de decisão de anúncios (ADS) ou o servidor de origem. Esses erros têm um código de resposta HTTP 5xx.

Tópicos

- [Erros de reprodução do cliente retornados por AWS Elemental MediaTailor](#)
- [Erros de reprodução do servidor retornados por AWS Elemental MediaTailor](#)
- [Exemplos de erros de reprodução](#)

Erros de reprodução do cliente retornados por AWS Elemental MediaTailor

Orientação geral:

- Você pode encontrar informações detalhadas sobre a maioria dos erros nos cabeçalhos e no corpo da resposta.
- Para alguns erros, você precisa verificar suas configurações. Você pode recuperar as configurações da sua configuração de reprodução em. AWS Elemental MediaTailor Para a API, o recurso é `GetPlaybackConfiguration/Name`. Para obter detalhes, consulte a [Referência da API do AWS Elemental MediaTailor](#).

A tabela a seguir lista os códigos de erro do cliente que são retornados pelas atividades de manipulação do manifesto AWS Elemental MediaTailor, as causas prováveis e as ações que você pode tomar para resolvê-las.

Código	Nome de exceção	Significado	O que fazer
400	<code>BadRequestException</code>	MediaTailor não consegue atender à solicitação devido a um ou mais erros na formatação ou no conteúdo. Um parâmetro pode ser formatado incorretamente, ou a solicitação pode conter uma configuração de reprodução ou ID de sessão inválido.	Verifique se a solicitação está formatada corretamente e se contém informações precisas. Verifique se a configuração do endpoint de reprodução no player corresponde à configuração <code>ManifestEndpointPrefix</code> retornada por <code>GetPlaybackConfiguration</code> . Tente sua solicitação novamente.
400	<code>AccessDeniedException</code>	O cabeçalho do host fornecido na solicitação não corresponde ao prefixo do endpoint do manifesto que está configurado no URL de reprodução do MediaTailor. A CDN pode estar configurada incorretamente.	Verifique as configurações da CDN e se você está usando o prefixo do endpoint do manifesto correto para o MediaTailor. Tente sua solicitação novamente.
400	<code>NotFoundException</code>	MediaTailor não consegue encontrar as informações especificadas. Os possíveis motivos incluem um URL que não está	Verifique a configuração e a validade da sua solicitação e, em seguida, reinicialize a sessão.

Código	Nome de exceção	Significado	O que fazer
		mapeado para nada no serviço, uma configuração que não está definida ou uma sessão que está indisponível.	
40	ConflictException	Um player tentou carregar várias listas de reprodução simultaneamente para uma única sessão. Como resultado, MediaTailor detectou um conflito de consistência na sessão. Esse problema ocorre para players HLS.	Verifique se o player solicita listas de reprodução, uma de cada vez. Isso está de acordo com a especificação HLS.
41	Gone	Um operador do AWS Support bloqueou a sessão de um jogador ou a configuração do cliente. AWS O Support faz isso em raras circunstâncias, quando detectamos um volume muito alto de solicitações 4xx provenientes de tráfego errôneo para uma única sessão ou configuração.	Se você acha que a solicitação não deve ser bloqueada, entre em contato com o AWS Support . Eles podem analisar e remover o filtro de bloqueio, se adequado.

Se precisar de mais ajuda, entre em contato com o [AWS Support](#).

Erros de reprodução do servidor retornados por AWS Elemental MediaTailor

Orientação geral:

- Você pode encontrar informações detalhadas sobre a maioria dos erros nos cabeçalhos e no corpo da resposta.
- Para alguns erros, você precisa verificar suas configurações. Você pode recuperar as configurações da sua configuração de reprodução em. AWS Elemental MediaTailor Para a API, o

recurso é `GetPlaybackConfiguration/Name`. Para obter detalhes, consulte a [Referência da API do AWS Elemental MediaTailor](#).

A tabela a seguir lista os códigos de erro do servidor retornados pelas atividades de manipulação do manifesto AWS Elemental MediaTailor, as causas prováveis e as ações que você pode tomar para resolvê-las.

Código	Nome de exceção	Significado	O que fazer
50	InternalServerError	Exceção não tratada.	Repetir a solicitação . Se o problema persistir, verifique a integridade relatada da MediaTailor sua AWS região em https://status.aws.amazon.com/ .
50	BadGatewayException	O endereço do servidor de origem ou o endereço do servidor de decisão de anúncios (ADS) são inválidos. Exemplos de endereços inválidos são um endereço IP privado e localhost .	Verifique se a configuração tem as definições corretas para o servidor de origem e o ADS e tente enviar a solicitação novamente.
50	UnsupportedManifestException	Ou o manifesto de origem foi alterado para que não seja possível personalizá-lo ou MediaTailor não é compatível com o formato do manifesto de origem.	Isso pode afetar somente a sessão individual. Reinicialize a sessão. Geralmente, você pode fazer isso ao atualizar a página no visualizador. Se o problema persistir, verifique se o MediaTailor é compatível com o formato do manifesto de origem. Para mais informações, consulte Integrando uma fonte de conteúdo .

Código	Nome de exceção	Significado	O que fazer
50	LoadShed	MediaTailor enfrentou uma restrição de recursos ao atender sua solicitação.	Repetir a solicitação . Se o problema persistir, verifique a integridade relatada da MediaTailor sua AWS região em https://status.aws.amazon.com/ .
50	ThrottlingException	Suas transações por segundo atingiram a cota, e o MediaTailor está limitando seu uso.	Repetir a solicitação . Você também pode verificar a saúde relatada da MediaTailor sua AWS região em https://status.aws.amazon.com/ . É possível aumentar a cota de transações por segundo. Para obter mais informações, consulte the section called “Cotas na inserção de anúncios” .
50	GatewayTimeoutException	Ocorreu um tempo limite ao MediaTailor entrar em contato com o servidor de origem.	Repetir a solicitação . Se o problema persistir, verifique a integridade do servidor de origem e se o servidor de origem está respondendo dentro do tempo limite do servidor de origem do conteúdo que está listado em the section called “Cotas na inserção de anúncios” .

Se precisar de mais ajuda, entre em contato com o [AWS Support](#).

Exemplos de erros de reprodução

Esta seção lista alguns exemplos dos erros de reprodução que você pode ver nas interações da linha de comando com AWS Elemental MediaTailor.

O exemplo a seguir mostra o resultado quando ocorre um tempo limite entre AWS Elemental MediaTailor o servidor de decisão de anúncios (ADS) ou o servidor de origem.

```
~[> 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: mediataylor.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]}%
```

Desduplicação de beacons

Esta seção fornece orientação para solução de problemas MediaTailor do recurso de desduplicação de beacon no rastreamento do lado do servidor.

Sintoma: Contagem de faróis mais baixa do que a observada anteriormente

Causa: A deduplicação de beacons está impedindo eventos duplicados que antes eram contados várias vezes.

Resolução: Esse é o comportamento esperado e indica uma medição mais precisa. As novas contagens representam eventos reais de visualização de anúncios em vez de solicitações de segmentos técnicos. Atualize suas linhas de base de relatórios para refletir as contagens deduplicadas.

Sintoma: a contagem de faróis varia entre diferentes implementações de jogadores

Causa: diferentes players podem ter diferentes padrões de solicitação de segmentos, mas a deduplicação garante contagens de beacons consistentes, independentemente dessas diferenças.

Resolução: Esse é o comportamento esperado. A deduplicação fornece uma medição consistente em diferentes tipos de players e condições de rede.

Sintoma: Discrepância entre as contagens de faróis esperadas e reais

Causa: As expectativas anteriores podem ter sido baseadas em contagens inflacionadas devido ao disparo duplicado do farol.

Resolução: revise os dados históricos para identificar padrões de beacons duplicados. Use as novas contagens deduplicadas como a linha de base precisa para medições e relatórios.

Solução de problemas de parâmetros

Para solucionar problemas relacionados a variáveis dinâmicas de anúncios, parâmetros de consulta de manifesto, restrições de caracteres, limitações de comprimento e aliases de configuração, consulte [MediaTailor guia de solução de problemas de parâmetros](#) em [MediaTailor variáveis dinâmicas de anúncios para solicitações de ADS](#).

MediaTailor guia de solução de problemas para ignorar anúncios

Ignorar anúncios é um dos problemas mais comuns relatados MediaTailor pelos clientes. Este guia ajuda você a identificar problemas de ignorar anúncios e fornece procedimentos de solução de problemas passo a passo.

Sintomas e impacto

Quando ocorre a omissão de anúncios, você pode observar os seguintes sintomas:

- Os anúncios não aparecem durante os intervalos publicitários esperados
- Os intervalos publicitários mostram conteúdo em vez de anúncios
- Reprodução inconsistente de anúncios em diferentes sessões de visualização
- CloudWatch registros mostrando AdSkipped eventos com vários motivos de salto

Ignorar anúncios afeta diretamente a receita e pode criar experiências ruins para os espectadores. Resolva esses problemas imediatamente para manter o desempenho ideal.

Identificando os motivos do salto

Use o CloudWatch Logs Insights para consultar o grupo de MediaTailor/AdDecisionServerInteractions registros para anúncios ignorados:

```
fields @timestamp, avail.availId, skippedAds.0.skippedReason,  
      skippedAds.0.creativeUniqueId  
| filter eventType = "FILLED_AVAIL" and ispresent(skippedAds.0.skippedReason)  
| sort @timestamp desc
```

Essa consulta retorna os eventos de salto de anúncios mais recentes com seus motivos específicos para ajudar você a identificar padrões.

Tópicos

- [MediaTailor Solução de problemas com NEW_CREATIVE e ignorando anúncios](#)
- [MediaTailor Solução de problemas de tempo limite e salto do ADS](#)
- [MediaTailor VAST analisando e ignorando a solução de problemas](#)
- [MediaTailor incompatibilidade de duração e solução de problemas de salto](#)
- [MediaTailor variável de sessão e ignorando a solução de problemas](#)
- [MediaTailor monitoramento e alertas de saltos de anúncios](#)
- [MediaTailor melhores práticas de prevenção de saltos de anúncios](#)

- [MediaTailor guia de referência para ignorar anúncios](#)

MediaTailor Solução de problemas com NEW_CREATIVE e ignorando anúncios

Quando os anúncios são ignorados com o NEW_CREATIVE motivo, AWS Elemental MediaTailor encontramos um anúncio que exige transcodificação antes da inserção. Este guia de solução de problemas explica as causas e fornece procedimentos de resolução passo a passo.

O que faz com que NEW_CREATIVE pule

MediaTailor transcodifica anúncios com base em três fatores principais: ID criativa, ID da conta da AWS e conjunto de variantes de transcodificação (as representações de reprodução do stream de conteúdo subjacente). Quando qualquer parte do conjunto de variantes da ID criativa ou da transcodificação é diferente, MediaTailor reconhece o anúncio como uma nova variante que requer transcodificação.

Alterações na taxa de bits

Alterações na taxa de bits podem fazer com que NEW_CREATIVE pule quando a taxa de bits não coincide após o arredondamento para os 8.000 bits mais próximos. Isso geralmente ocorre nas seguintes situações:

- O fluxo de conteúdo subjacente altera o manifesto principal
- Novas sessões são criadas com variantes diferentes das sessões existentes
- As taxas de bits do fluxo de conteúdo são inconsistentes nas sessões de reprodução

Colete a variação da largura de banda do trabalho

Para trabalhos de colheita, MediaTailor usa um limite de variação de largura de banda de 15% ao combinar anúncios com fluxos de conteúdo. Se houver uma diferença de 15% ou mais entre a largura de banda da transmissão ao vivo e a largura de banda do arquivo HLS coletado, o anúncio será transcodificado novamente. Isso pode fazer com que NEW_CREATIVE seja ignorado durante o processo de transcodificação.

Esse comportamento geralmente ocorre quando:

- Os mesmos anúncios pré-transcodificados são usados para streaming ao vivo e conteúdo VOD

- O conteúdo coletado tem parâmetros de codificação diferentes dos da transmissão ao vivo original
- As diferenças de largura de banda excedem o limite de 15% entre o conteúdo ao vivo e o conteúdo coletado

Conflitos de ID criativa

Quando MediaTailor encontra uma ID criativa diferente para um arquivo de mídia que já foi transcodificado, ocorre a seguinte sequência:

1. O anúncio é ignorado pelo motivo `NEW_CREATIVE`
2. Isso leva a uma tentativa desnecessária de transcodificação
3. O criativo está marcado como `DUPLICATE_TRANSCODE` ou `COPY_DEDUP`

Note

MediaTailor não expira nem exclui anúncios transcodificados. Nós os armazenamos em um bucket MediaTailor-owned S3 indefinidamente.

Etapas de resolução

Para resolver problemas de ignorar anúncios do `NEW_CREATIVE`:

1. Verifique se seu servidor de decisão de anúncio retorna IDs criativas consistentes para o mesmo conteúdo do anúncio.
2. Verifique se seu stream de conteúdo mantém taxas de bits e conjuntos de variantes consistentes.
3. Considere implementar a pré-busca de anúncios para garantir que os anúncios sejam transcodificados antes da reprodução. Para obter mais informações, consulte [the section called “Pré-busca de anúncios”](#).
4. Para problemas persistentes, entre em contato com o [AWS Support](#) para obter assistência adicional na solução de problemas.

Monitorando padrões `NEW_CREATIVE`

Use essa consulta do CloudWatch Logs Insights para analisar padrões na omissão de anúncios `NEW_CREATIVE`:

```
fields @timestamp, sessionId, creativeId, skipReason, MediaFileSourceUrl
| filter skipReason = "NEW_CREATIVE"
| stats count() by creativeId, MediaFileSourceUrl
| sort count desc
| limit 50
```

MediaTailor Solução de problemas de tempo limite e salto do ADS

Quando os anúncios são ignorados por motivos relacionados ADS_TIMEOUT ou relacionados, você tem problemas de conectividade ou desempenho com seu servidor de decisão de anúncios. AWS Elemental MediaTailor requer comunicação confiável com seu ADS para inserir anúncios com sucesso. Este guia de solução de problemas explica como identificar e resolver esses problemas de conectividade.

Problemas comuns de conectividade do ADS

Os problemas comuns de conectividade do ADS incluem o seguinte:

- Servidor de decisão de anúncios não acessível a partir de MediaTailor
- O ADS não está respondendo dentro do período de tempo limite configurado
- O ADS não consegue lidar com o volume de solicitações durante os períodos de pico
- Problemas de conectividade de rede entre MediaTailor e seu ADS

Etapas de resolução

Para resolver problemas de tempo limite do ADS:

1. Verifique se seu servidor de decisão de anúncios está acessível a partir de MediaTailor.
2. Verifique se seu ADS responde dentro do período de tempo limite configurado.
3. Certifique-se de que seu ADS possa lidar com o volume de solicitações durante os períodos de pico.
4. Considere implementar uma estratégia de anúncios alternativos para quando seu ADS principal estiver indisponível.

Monitorando o desempenho do AD

Configure CloudWatch alarmes para a `AdDecisionServer.Timeouts` métrica para detectar proativamente problemas de conectividade do ADS.

MediaTailor VAST analisando e ignorando a solução de problemas

Quando os anúncios são ignorados com `VAST_PARSING_ERROR` ou `MEDIA_FILE_UNAVAILABLE`, você tem problemas com suas respostas VAST ou arquivos de mídia publicitária. AWS Elemental MediaTailor requer respostas VAST formatadas adequadamente e arquivos de mídia acessíveis para uma inserção bem-sucedida do anúncio. Este guia de solução de problemas explica como identificar e resolver esses problemas.

Problemas comuns do VAST

Os problemas comuns do VAST incluem o seguinte:

- Formato de resposta VAST não compatível com a especificação VAST
- URLs de arquivos de mídia na resposta VAST não acessíveis publicamente
- Caracteres especiais não codificados corretamente em VAST XML
- Resposta VAST sem arquivos ou formatos de mídia necessários

Etapas de resolução

Para resolver problemas de análise do VAST:

1. Valide seu formato de resposta VAST em relação à especificação VAST.
2. Certifique-se de que todos os URLs de arquivos de mídia na resposta VAST estejam acessíveis ao público.
3. Verifique a codificação adequada de caracteres especiais em seu XML VAST.
4. Verifique se sua resposta VAST inclui arquivos de mídia em formatos compatíveis com MediaTailor.

Solução de problemas do VAST wrapper

Para problemas com `INVALID_VAST_WRAPPER_AD` ou `REJECTED_REPLICA_VAST`:

- Valide as respostas do invólucro VAST em relação à especificação VAST

- Certifique-se de que todos os elementos do invólucro estejam formatados corretamente e contendo elementos válidos VASTAdTagURI
- Verifique a configuração do servidor de anúncios para políticas de detecção de conteúdo duplicado
- Garanta que as respostas do VAST contendo conteúdo criativo exclusivo na mesma sessão

MediaTailor incompatibilidade de duração e solução de problemas de salto

Quando os anúncios são ignorados por motivos relacionados ao formato ou por motivos relacionados ao formato, você tem problemas de compatibilidade de duração ou formato.

AVAIL_DURATION_EXCEEDED LEFTOVER_AVAIL_EXCEEDED_THRESHOLD AWS Elemental MediaTailor requer formatação e correspondência adequadas da duração entre o conteúdo do anúncio e o tempo de intervalo disponível para o anúncio. Este guia de solução de problemas explica como identificar e resolver esses problemas relacionados à duração.

Problemas comuns de duração

Os problemas comuns de duração incluem o seguinte:

- Anúncios maiores do que a duração disponível do intervalo publicitário
- Marcadores de intervalo de anúncio incorretos no conteúdo, sinalizando duração incorreta
- Limite de personalização não atingido para inserção de anúncios
- Formato de duração incorreto nas EXT-X-CUE-OUT tags

Requisitos de formato de duração

O parâmetro de duração da EXT-X-CUE-OUT tag deve ser formatado como um valor inteiro, não como um formato de duração ISO 8601.

Requisitos de formato de duração

Formato	Exemplo	Status
Inteiro (correto)	32	Suportado - representa 32 segundos
Decimal (correto)	30.000	Suportado - representa 30 segundos
ISO 8601 (incorreto)	PT32S	Não suportado - causa falhas de inserção

Etapas de resolução

Para resolver problemas de incompatibilidade de duração:

1. Garanta que seu ADS retorne anúncios que se encaixem na duração do intervalo publicitário disponível.
2. Verifique se os marcadores de quebra de anúncio no conteúdo sinalizam corretamente a duração pretendida.
3. Verifique se os parâmetros de EXT-X-CUE-OUT duração usam o formato inteiro.
4. Considere ajustar o limite de personalização, se apropriado para seu caso de uso.

MediaTailor variável de sessão e ignorando a solução de problemas

As variáveis de sessão desempenham um papel fundamental na segmentação e seleção de anúncios em AWS Elemental MediaTailor. A configuração incorreta da variável de sessão é uma causa comum de problemas de ignorar anúncios. Este guia abrangente de solução de problemas explica como identificar e resolver problemas de variáveis de sessão que podem impedir a inserção bem-sucedida de anúncios.

Problemas comuns de variáveis de sessão

Os problemas comuns de variáveis de sessão incluem o seguinte:

- Variáveis obrigatórias ausentes: seu servidor de decisão de anúncios pode exigir variáveis específicas que não estão sendo fornecidas.
- Sintaxe de variável incorreta: as variáveis devem usar a sintaxe correta (por exemplo, `[session.id]` em vez de `${session.id}`).
- Problemas de codificação de URL: caracteres especiais em valores variáveis podem precisar de uma codificação de URL adequada.
- Parâmetros inconsistentes do jogador: os parâmetros do jogador devem ser transmitidos de forma consistente entre as sessões.
- Falhas na resolução dinâmica de variáveis: variáveis que não podem ser resolvidas são substituídas por sequências vazias.
- SCTE-35 Problemas de análise do UPID: problemas com a segmentação do processamento do UPID podem causar falhas na resolução da variável da sessão.

Verificando a resolução variável da sessão

Para verificar se as variáveis da sessão estão sendo resolvidas adequadamente:

1. Ative o registro de depuração para sua configuração MediaTailor
2. Verifique o grupo de MediaTailor/AdDecisionServerInteractions registros para ver os URLs reais de solicitação do ADS
3. Verifique se todas as variáveis no URL do modelo foram substituídas por valores apropriados
4. Procure todas as variáveis que foram substituídas por strings vazias, o que pode indicar falhas na resolução

SCTE-35 Solução de problemas de análise do UPID

Problemas com o processamento de UPID de SCTE-35 segmentação podem causar problemas com variáveis de sessão:

- Requisitos de formato: o UPID deve ter `segmentation_upid_type` 12 e incluir `format_identifier` para um processamento adequado
- Regras de análise: o UPID decodificado pode conter delimitadores de dois pontos para vários valores. O número de variáveis de modelo e tokens UPID decodificados deve ser igual
- Formatos inválidos: evite dois pontos duplos sem valores (por exemplo, `: :` ou `:46175218: :4053`), pois eles causam falhas na análise

MediaTailor monitoramento e alertas de saltos de anúncios

O monitoramento proativo ajuda você a detectar e resolver problemas de ignorar anúncios antes que eles afetem significativamente sua receita. AWS Elemental MediaTailor fornece métricas abrangentes e recursos de registro que permitem o monitoramento eficaz do desempenho da inserção de anúncios. Este guia explica como configurar um monitoramento eficaz para problemas de ignorar anúncios.

Principais CloudWatch métricas a serem monitoradas

Configure CloudWatch alarmes para essas MediaTailor métricas principais:

- `AdDecisionServer.Ads.Skipped`- Contagem de anúncios ignorados

- `AdDecisionServer.Timeouts`- Contagem de tempos limite do ADS
- `Avail.FilledDuration`- Duração dos intervalos publicitários preenchidos
- `Avail.SlateOnly`- Contagem de intervalos publicitários preenchidos apenas com ardósia

Consultas avançadas CloudWatch do Logs Insights

Use essas consultas especializadas para solucionar problemas detalhados:

Análise abrangente da sessão

Para uma análise detalhada do comportamento de inserção de anúncios em uma sessão específica:

```
fields @timestamp, sessionId, eventType, creativeId, skipReason, adBreakIndex
| filter sessionId = "your-session-id-here"
| filter eventType in ["FILLED_AVAIL", "SKIPPED_AVAIL", "MAKING_ADS_REQUEST"]
| sort @timestamp asc
| limit 100
```

Encontrando IDs criativos

Para identificar as Creative IDs a partir de eventos FILLED_AVAIL:

```
fields @timestamp, sessionId, eventType
| filter sessionId like /sessionId/ and eventType!='BEACON_FIRED'
| sort @timestamp desc
```

Note

`sessionId` substitua pelo ID real da sessão que você está investigando.

MediaTailor melhores práticas de prevenção de saltos de anúncios

A implementação dessas práticas recomendadas ajuda a evitar problemas de ignorar anúncios antes que eles ocorram, garantindo melhor desempenho de inserção de anúncios e proteção de receita

com. AWS Elemental MediaTailor Essas medidas proativas abordam as causas mais comuns de pular anúncios e ajudam a manter a exibição consistente de anúncios.

Medidas proativas

- Implemente a pré-busca de anúncios: use o recurso MediaTailor de pré-busca do usuário para garantir que os anúncios sejam transcodificados antes da reprodução. Consulte [the section called “Pré-busca de anúncios”](#) para obter detalhes de implementação
- Mantenha IDs criativos consistentes: garanta que seu servidor de decisão de anúncios use IDs criativos consistentes para o mesmo conteúdo de anúncio em todas as sessões
- Garanta a formatação adequada da duração: use valores inteiros para os parâmetros de EXT-X-CUE-OUT duração em vez do formato ISO 8601
- Configure a otimização de VOD: defina o máximo ConcurrentAdsRequests para fluxos de VOD com vários intervalos publicitários para reduzir a carga do servidor
- Otimize o desempenho do ADS: configure seu servidor de decisão de anúncios para responder rapidamente e lidar com volumes de pico de tráfego

Diretrizes de implementação

- Implemente o tratamento adequado de erros: configure o conteúdo do Slate para preencher intervalos publicitários quando os anúncios não puderem ser inseridos
- Teste minuciosamente: valide seu fluxo de trabalho de inserção de anúncios em diferentes dispositivos e condições de rede
- Implemente estratégias alternativas: configure fontes de anúncios de backup ou anúncios padrão para quando as fontes de anúncios primárias falharem
- Monitore padrões de transcodificação: monitore os CloudWatch registros para verificar a eficiência da transcodificação e entre em contato com o AWS [Support](#) se você observar padrões que indiquem problemas de transcodificação

MediaTailor guia de referência para ignorar anúncios

Esta seção fornece informações de referência abrangentes sobre os motivos do abandono do anúncio e links para a documentação relacionada do. AWS Elemental MediaTailor Use este guia de referência para entender os significados específicos dos diferentes motivos de omissão e encontrar recursos adicionais para solução de problemas.

Referência completa dos motivos para ignorar o anúncio

MediaTailor registra os motivos específicos pelos quais os anúncios são ignorados na mensagem de registro de FILLED_AVAIL eventos do grupo de MediaTailor/AdDecisionServerInteractions registros.

Motivos preenchidos e ignorados

Ignorar o motivo	Description
NEW_CREATIVE	O anúncio ainda não foi transcodificado. Isso ocorre quando se depara com MediaTailor um novo anúncio criativo que requer transcodificação antes da inserção.
PROFILE_NOT_FOUND	O perfil de MediaConvert transcodificação associado à configuração da sessão não existe, impedindo a preparação do anúncio.
TRANSCODE_ERROR	O processo de transcodificação do anúncio encontrou um erro e não foi concluído.
TRANSCODE_IN_PROGRESS	A transcodificação do anúncio ainda está em andamento e ainda não está pronta para inserção.
INTERNAL_ERROR	Ocorreu um MediaTailor erro interno ao manipular o anúncio, impedindo a inserção.
AVAIL_DURATION_EXCEEDED	O anúncio não se encaixa na duração restante do intervalo publicitário.
LEFTOVER_AVAIL_EXCEEDED_THRESHOLD	A duração cumulativa de todos os anúncios que poderiam ter sido inseridos não atende ao limite de personalização configurado para a sessão.
VAST_PARSING_ERROR	A resposta VAST do servidor de decisão do anúncio contém erros ou está mal formada.
ADS_TIMEOUT	O servidor de decisão do anúncio não respondeu dentro do período de tempo limite configurado.

Ignorar o motivo	Description
MEDIA_FILE_UNAVAILABLE	Os arquivos de mídia publicitária especificados na resposta VAST não estão acessíveis.
SESSION_INITIALIZATION_FAILED	A MediaTailor sessão falhou ao inicializar corretamente, geralmente devido a variáveis de sessão incorretas.
EARLY_CUE_IN	O intervalo publicitário terminou mais cedo do que o esperado devido a um sinal de alerta antecipado, impedindo que o anúncio fosse totalmente inserido.
NO_VARIANT_MATCH	O anúncio criativo não tem uma variante que corresponda aos parâmetros de codificação do fluxo de conteúdo (taxa de bits, resolução, codec).
NO_MODEL_CREATIVE_MATCH	O anúncio criativo não corresponde aos requisitos de modelo ou formato esperados para a configuração de reprodução atual.
REJECTED_REPLICA_VAST	A resposta do VAST foi rejeitada devido às políticas de detecção de conteúdo duplicado ou de réplicas.
INVALID_VAST_WRAPPER_AD	O anúncio de invólucro VAST contém elementos de embalagem inválidos ou malformados que impedem a inserção bem-sucedida do anúncio.
IMPORT_ERROR	Ocorreu um erro durante o processo de importação do anúncio, impedindo que o anúncio fosse processado para inserção.
IMPORT_IN_PROGRESS	O processo de importação de anúncios está em andamento e ainda não foi concluído.

Recursos relacionados

Para obter mais informações sobre como solucionar problemas de omissão de anúncios, consulte estes tópicos relacionados:

- [the section called “Variáveis de anúncios dinâmicos”](#) - Guia abrangente sobre variáveis dinâmicas de anúncios em MediaTailor

- [the section called “Pré-busca de anúncios”](#) - Como implementar a pré-busca de anúncios para evitar o salto relacionado à transcodificação
- [the section called “Monitoramento com CloudWatch métricas”](#) - Monitoramento MediaTailor com CloudWatch métricas
- [the section called “Visualizar logs”](#) - Como visualizar e analisar MediaTailor registros
- [the section called “Solução de problemas de fluxo de eventos”](#) - Entendendo o fluxo de eventos de inserção de anúncios

AWS Elemental MediaTailor recursos

A tabela a seguir lista os recursos relacionados que serão úteis à medida que você trabalhar com o AWS Elemental MediaTailor.

Recurso	Description
Padrão SCTE: SCTE 35	O documento padrão SCTE para SCTE35.
Aulas e workshops	Links para cursos especializados e baseados em funções, bem como laboratórios individualizados para ajudar a aprimorar suas AWS habilidades e ganhar experiência prática.
AWS ferramentas para desenvolvedores	Links para ferramentas de desenvolvedor SDKs, kits de ferramentas de IDE e ferramentas de linha de comando para desenvolver e gerenciar AWS aplicativos.
AWS whitepapers	Links para uma lista abrangente de AWS white papers técnicos, abrangendo tópicos como arquitetura, segurança e economia, escritos por arquitetos de AWS soluções ou outros especialistas técnicos.
AWS Support center	O hub para criar e gerenciar seus casos de AWS Support. Também inclui links para outros recursos úteis, como fóruns, informações técnicas FAQs, status de integridade do serviço e AWS Trusted Advisor.
AWS Suporte	A principal página da web para obter informações sobre o AWS Support one-on-one, um canal de suporte de resposta rápida para ajudá-lo a criar e executar aplicativos na nuvem.

Recurso	Description
Entre em contato conosco	Um ponto de contato central para consultas sobre AWS cobrança, conta, eventos, abuso e outros problemas.
AWS termos do site	Informações detalhadas sobre nossos direitos autorais e marca registrada; conta, licença e acesso ao site, entre outros tópicos.

Histórico do documento para AWS Elemental MediaTailor

A tabela a seguir descreve alterações importantes feitas nesta documentação.

Alteração	Descrição	Data
<u>Account-level dimensões por motivos de ignoração do anúncio</u>	Foram adicionadas emissões em nível de conta (sem dimensões) para as SkippedReason métricas.	21 de julho de 2026
<u>Novas dimensões métricas e em nível de conta</u>	A Avail.Complete métrica foi adicionada. Emissões adicionadas em nível de conta (sem dimensões) para as Requests métricas AdsBilled, Avail.Impression, Avail.Complete, Avail.FilledDuration, Avail.Duration, e AdDecisionServer.Ads.	30 de junho de 2026
<u>Autenticação automática do lado do servidor com as plataformas de anúncios do Google</u>	O capítulo foi atualizado para refletir que MediaTailor agora autentica automaticamente as conexões de servidor a servidor com o Google Ad Manager (GAM), o Google Campaign Manager (GCM) e o Google Display & Video 360 (DV360). Nenhuma ação do cliente ou ticket de suporte é necessária. A seção do lado do servidor foi atualizada para descrever URI-based as	15 de abril de 2026

regras MediaTailor aplicáveis e removeu o procedimento manual de tíquetes de suporte.

[Modelagem de tráfego TPS adicionada](#)

Seções atualizadas para incluir informações sobre a limitação de solicitações ao ADS com base no TPS.

2 de outubro de 2025

[prefetchScheduleName adicionado](#)

Foram adicionadas informações sobre prefetchScheduleName nos registros do ADS.

24 de setembro de 2025

[Guia de solução de problemas com anúncios reestruturados](#)

Guia de solução de problemas reestruturado para focar cada seção em um trabalho específico a ser feito.

26 de agosto de 2025

[Documentação de manipulação de parâmetros de consulta adicionada](#)

Foi adicionada uma seção explicando como MediaTailor manipula os parâmetros de consulta para funcionalidades específicas da origem, como visualização com mudança de horário com. MediaPackage

26 de agosto de 2025

[Guia de configuração do SGAI VOD](#)

Foi adicionada orientação de configuração para inserção de anúncios guiada pelo servidor com conteúdo de vídeo sob demanda.

7 de agosto de 2025

[Orientação integrada de solução de problemas de parâmetros](#)

Foi adicionada uma seção abrangente de solução de problemas que abrange erros de restrição de caracteres, problemas de limitação de comprimento, problemas de alias de configuração, processamento de parâmetros, depuração de fluxo e melhores práticas de segurança.

4 de agosto de 2025

[Documentação aprimorada de formatação de parâmetros](#)

Foram adicionadas restrições abrangentes de caracteres, URL-encoding suporte e limitações de comprimento para parâmetros de consulta de manifesto e parâmetros ADS.

1.º de agosto de 2025

[Documentação de parâmetros aprimorada com referência abrangente](#)

Foram adicionadas restrições abrangentes de caracteres de parâmetros, limitações de comprimento, URL-encoding suporte, documentação de aliases de configuração e métodos de inicialização de sessão com base na análise do AWS Guide.

1.º de agosto de 2025

[Documentação aprimorada de parâmetros de consulta de manifesto](#)

Foi adicionado um comportamento abrangente de posicionamento de parâmetros, restrições de caracteres, URL-encoding suporte e diferenças específicas de protocolos com base na análise do AWS Guide.

1.º de agosto de 2025

[Documentação aprimorada de variáveis de domínio](#)

Foi adicionada documentação abrangente de aliases de configuração, campos compatíveis e regras de parâmetros em nível de domínio com base na análise do AWS Guide.

1.º de agosto de 2025

[Orientação adicional para solução de problemas de fluxo de eventos](#)

Nova seção que fornece orientação prática para usar a análise de fluxo de MediaTailor eventos para diagnosticar e resolver problemas de inserção de anúncios.

31 de julho de 2025

[Documentação do fluxo de eventos de inserção de anúncios adicionada](#)

Nova seção que explica o fluxo cronológico de MediaTailor eventos durante a personalização do manifesto para ajudar os usuários a entender o fluxo de trabalho.

23 de julho de 2025

Documentação aprimorada do fluxo de trabalho de rastreamento do lado do cliente	Foi adicionada documentação abrangente do fluxo de trabalho de relatórios do lado do cliente com diagrama de sequência detalhado e explicação passo a passo do processo.	16 de julho de 2025
Documentação de integração CDN	Foram adicionadas informações mais abrangentes sobre a configuração de uma CDN com MediaTailor.	30 de junho de 2025
Integração CDN adicionada para montagem de canais	Foi adicionada documentação para integrar uma CDN com o Channel Assembly para otimizar a entrega de conteúdo.	16 de maio de 2025
Tipos de playlist HLS	Definições adicionadas para listas de reprodução multivariantes e de mídia HLS.	9 de maio de 2025
Tipos de manifesto DASH	Definições adicionadas para componentes do manifesto DASH.	9 de maio de 2025
Informações sobre o tipo de playlist HLS adicionadas	Foram adicionadas explicações sobre playlists multivariantes e variantes para contextualizar os exemplos.	9 de maio de 2025
Informações sobre o tipo de manifesto DASH adicionadas	Foram adicionadas explicações sobre a estrutura e os componentes do MPD para fornecer contexto para os exemplos.	9 de maio de 2025

Pré-busca recorrente adicionada	Seções atualizadas para incluir informações sobre agendamentos de pré-busca recorrentes.	28 de abril de 2025
Informações de filtragem de log por configuração adicionadas	Foram adicionadas informações sobre a filtragem de registros MediaTailor emitidos para uma configuração.	31 de março de 2025
Foram adicionadas informações de registros vendidos	Foi adicionada uma nova seção sobre o uso de registros vendidos para publicar registros que MediaTailor emitem..	4 de fevereiro de 2025
Visão geral de anúncios pré-condicionados adicionada	Foi adicionada uma nova seção que descreve como MediaTailor gerencia a inserção de anúncios quando você usa anúncios pré-condicionados.	30 de janeiro de 2025
Adicionado e condicionamento	Foi adicionada a configuração de condicionamento do arquivo de mídia de streaming.	30 de janeiro de 2025
Guia de configuração ao vivo do SGA	Diretrizes de configuração adicionadas para inserção de anúncios guiados pelo servidor com conteúdo ao vivo.	15 de janeiro de 2025
Tipos de registro adicionados	Foi adicionada uma nova seção que lista os tipos de log MediaTailor emitidos.	15 de janeiro de 2025

Informações de filtragem de registros adicionadas	Foi adicionada uma nova seção sobre a filtragem de registros que MediaTailor emitem.	15 de janeiro de 2025
Foi adicionada a formatação de parâmetros de consulta suportada	Seções adicionadas para formatação de parâmetros de consulta de manifesto e de parâmetros de consulta ADS.	2 de janeiro de 2025
Informações de integração para o Google Ad Manager	Foi adicionada uma seção sobre a integração MediaTailor com o Google Ad Manager do lado do servidor e do lado do cliente.	25 de novembro de 2024
Variável atualizada	Alterado breakabilityStartTime paraavailabilityStartTime .	6 de maio de 2024
Detalhes de personalização adicionados	Foram adicionados os detalhes de Insertion Mode personalização.	6 de maio de 2024
Regras do programa	Foi adicionado novo conteúdo sobre as regras do programa.	25 de abril de 2024
Conteúdo atualizado sobre a criação de canais	Foram adicionadas informações sobre as regras do programa ao criar canais.	20 de abril de 2024
Conteúdo atualizado sobre a adição de programas	Foram adicionadas informações sobre as regras do programa ao adicionar programas.	20 de abril de 2024

Monitor de fluxo de trabalho	Analise os serviços de AWS mídia e crie mapas de sinais, visualizações do fluxo de trabalho de mídia, entre esses serviços. Use os mapas de sinal para gerar alarmes e notificações de monitoramento usando CloudWatch EventBridge, e. CloudFormation	11 de abril de 2024
AlternateMedia e o As Run Log	Foi adicionada uma observação sobre como AlternateMedia afeta o As Run Log.	28 de fevereiro de 2024
Configurações atualizadas do manifesto	Foram adicionadas informações sobre tags de passagem, dependendo do tipo de marcação de anúncio escolhido.	28 de novembro de 2023
SCTE-35 mensagens para pausas publicitárias	Foram adicionadas informações sobre SCTE-35 tags inseridas Daterange versus tipos de marcação de Scte35 Enhanced anúncios.	28 de novembro de 2023
Key-value pares para o tipo Enhanced Scte35 de marcação de anúncio	Foram adicionadas informações sobre como MediaTailor lida com pares de valores-chave enviados para o tipo de marcação de Enhanced Scte35 anúncio.	28 de novembro de 2023
Fontes e oportunidades de VOD	MediaTailor Agora eu posso detectar automaticamente oportunidades de anúncios em fontes de VOD.	6 de outubro de 2023

Novo tipo de autenticação SigV4 com detecção automática	MediaTailor agora suporta o tipo de AUTODETECT_SIGV4 acesso.	18 de agosto de 2023
Conteúdo atualizado de rastreamento do lado do cliente	Atualizou o conteúdo de rastreamento do lado do cliente para incluir informações adicionais.	12 de agosto de 2023
Configuração MediaTailor e MediaPackage fluxos de trabalho para usar fontes ativas	Foram adicionadas informações sobre configuração, requisitos gerais e comportamento quando MediaTailor os AWS Elemental MediaPackage fluxos de trabalho usam fontes ativas.	24 de maio de 2023
Documentação dos parâmetros de consulta do manifesto	Foi adicionada uma seção que descreve os parâmetros de consulta do manifesto.	26 de abril de 2023
Documentação de anúncios sobrepostos	Foi adicionada uma seção que descreve anúncios sobrepostos.	24 de abril de 2023
Decoração de ID de anúncio na documentação de manifestos	Foi adicionada uma seção que descreve a decoração do ID de anúncio nos manifestos.	24 de abril de 2023
Adicionado o modo de supressão AFTER_LIVE_EDGE	AFTER_LIVE_EDGE O modo de supressão de anúncios agora está disponível além do BEFORE_LIVE_EDGE modo 2.	21 de fevereiro de 2023
Novo registro de As Run	Novo tópico no registro As Run.	19 de janeiro de 2023

[Atualizações das melhores práticas do IAM](#)

Guia atualizado para alinhamento com as práticas recomendadas do IAM. Para saber mais, consulte [Práticas recomendadas de segurança no IAM](#).

27 de dezembro de 2022

[Atualizações das melhores práticas do IAM](#)

Guia atualizado para alinhamento com as práticas recomendadas do IAM. Para saber mais, consulte [Práticas recomendadas de segurança no IAM](#).

27 de dezembro de 2022

[Novas variáveis de dados privados do UPID de segmentação](#)

MediaTailor agora oferece suporte a variáveis de sessão adicionais `scte.segmentation_upid.private_data.{index}` que dividem dados privados delimitados por dois pontos da segmentação UPID tipo 12 (MPU).

22 de dezembro de 2022

[Conteúdo de cotas atualizado](#)

Informações de cotas atualizadas e reorganizadas.

13 de setembro de 2022

[Cotas de montagem de canais adicionadas e corrigidas](#)

Foram adicionadas cotas para fontes ativas, configurações de entrega de segmentos, solicitações de manifesto e transações de canais no serviço de montagem de MediaTailor canais.

13 de setembro de 2022

Cotas de montagem de canais adicionadas e corrigidas	Foram adicionadas cotas para registro, canal, fontes ao vivo, programas, locais de origem e políticas de canais no serviço de montagem de MediaTailor canais.	11 de setembro de 2022
Nova tabela de alertas de montagem de canais	Agora você pode ver uma tabela que explica os alertas de montagem de canais.	1º de setembro de 2022
Novas CloudWatch métricas da Amazon	Novas CloudWatch métricas adicionadas.	26 de junho de 2022
Tópico de chamadas publicitárias	Foi adicionado um link para suporte de pré-busca para respostas VAST.	25 de maio de 2022

[Novas variáveis de solicitação do ADS](#)

MediaTailor agora oferece suporte a essas SCTE-35 variáveis adicionais nas solicitações do ADS:

```
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.devi
ce_restri
ctions ,scte.no_r
egional_blackout_f
lag ,, scte.arch
ive_allowed_flag
scte.segmentation_
event_id e.
```

27 de abril de 2022

[Novo tópico de política gerenciada do IAM](#)

Foram adicionadas duas novas políticas gerenciadas para MediaTailor.

24 de novembro de 2021

[Nova política gerenciada pela AWSElementalMediaTailorReadOnly](#)

Foi adicionada uma nova política AWS gerenciada que concede permissões que permitem acesso somente para MediaTailor leitura aos recursos.

10 de novembro de 2021

<u>Nova política gerenciada pela AWSElementalMediaTailorFullAccess</u>	Foi adicionada uma nova política AWS gerenciada que permite acesso total aos MediaTailor recursos.	10 de novembro de 2021
<u>Novo tópico confuso para deputados</u>	Foi adicionado um tópico que explica como evitar o confuso problema do deputado.	4 de novembro de 2021
<u>Tópico de pré-busca de anúncios</u>	MediaTailor Agora, é possível pré-buscar anúncios para intervalos publicitários antes que eles ocorram.	12 de outubro de 2021
<u>Configurações de registro adicionadas para configurações de reprodução</u>	Use as configurações de registro para controlar as configurações relacionadas aos registros de configuração de reprodução.	28 de setembro de 2021
<u>SCTE-35 mensagens para pausas publicitárias</u>	Foram adicionadas informações sobre o uso de <code>segmentation_descriptor</code> mensagens junto com <code>time_signal</code> mensagens.	1º de setembro de 2021
<u>Novo modo de reprodução linear</u>	Foi adicionado um novo modo de reprodução linear.	1º de setembro de 2021
<u>Novo tipo de transição absoluta</u>	Foi adicionado suporte para tipos de transição absolutos , que você pode usar para definir a hora de início de um relógio de parede para seu programa em canais lineares.	1º de setembro de 2021

<u>Novo tópico de alertas de montagem de canais</u>	Agora você pode monitorar seus recursos de montagem de canais usando MediaTailor ou alertas. Quando ocorre um problema ou um possível problema com os recursos de montagem do canal, MediaTailor gera alertas.	14 de julho de 2021
<u>Cotas de montagem de canais corrigidas para solicitações de saída de canais</u>	Cotas corrigidas para solicitações de saída de canais no serviço de montagem de MediaTailor canais.	29 de junho de 2021
<u>Novo tipo de autenticação do local de origem</u>	MediaTailor agora oferece suporte à autenticação por token de acesso do Secrets Manager.	16 de junho de 2021
<u>Informações sobre o novo nível</u>	Foram adicionadas informações sobre os modos e os tipos de fonte que cada camada suporta.	13 de junho de 2021
<u>Novas informações sobre o tipo de fonte</u>	Para canais padrão, foram adicionadas informações sobre o tipo de fonte que o programa reproduz.	13 de junho de 2021
<u>Nova documentação de fontes MediaTailor ao vivo</u>	Uma fonte ao vivo representa uma única transmissão ao vivo que você adiciona ao local de origem. Depois de criar seu canal, você pode adicionar fontes ao vivo ao local de origem e associar cada fonte ao vivo a um programa.	13 de junho de 2021

Support para tipos adicionais de UPID	MediaTailor agora suporta os tipos de UPID de segmentação ADS Information (0xE) e Definida pelo Usuário (0x1).	15 de abril de 2021
Novas variáveis dinâmicas UPID de segmentação	Há três novas variáveis dinâmicas: <code>scte.segmentation_upid.assetId</code> , <code>scte.segmentation_upid.cueData.key</code> , <code>scte.segmentation_upid.cueData.value</code> e. Essas variáveis são usadas em conjunto com o tipo UPID de segmentação MPU (0xC) para fluxos de trabalho de podbuster.	15 de abril de 2021
Descrição do novo serviço de montagem de canais	Foram adicionadas informações sobre o novo serviço de montagem de canais.	11 de março de 2021
Nova documentação do serviço de montagem de MediaTailor canais	A montagem de canais é um novo serviço somente para manifestos que permite criar canais de streaming lineares usando seu conteúdo de vídeo sob demanda (VOD) existente.	11 de março de 2021
Adicionadas cotas de montagem de canais	Foram adicionadas cotas para o novo serviço de montagem de MediaTailor canais.	11 de março de 2021
Novos termos de montagem de canais	Termos adicionados que correspondem ao novo serviço de montagem de canais.	10 de março de 2021

<u>Suporte para marcação de montagem de canais</u>	Foi adicionado suporte para marcação de recursos de montagem de canais em AWS Elemental MediaTailor. Canais SourceLocations e marcação de VodSources suporte.	9 de março de 2021
<u>Novo tópico sobre variáveis dinâmicas</u>	MediaTailor agora oferece suporte a variáveis dinâmicas de domínio.	25 de fevereiro de 2021
<u>Foram adicionadas configurações opcionais de alias</u>	Use aliases de configuração junto com variáveis de domínio para configurar domínios dinamicamente durante a inicialização da sessão.	25 de fevereiro de 2021
<u>Nova variável de anúncio <code>scte.segmentation_upid</code> dinâmico</u>	Foi adicionado suporte para a variável de anúncio dinâmico de dados da <code>scte.segmentation_upid</code> sessão.	5 de dezembro de 2020
<u>Novo tópico de passagem de marcadores de anúncios</u>	A transmissão de marcadores de anúncios agora está disponível para manifestos HLS.	29 de outubro de 2020
<u>Configuração atualizada (configurações avançadas)</u>	A passagem do marcador de anúncio é uma nova configuração avançada de reprodução.	14 de outubro de 2020
<u>Novo modo de registro de depuração</u>	Novo tópico sobre o modo de registro DEBUG.	14 de agosto de 2020

Esclarecimento sobre o atributo de EXT-X-CUE-OUT duração para pára-choques	Os requisitos de bumpers foram atualizados para que, para o HLS, o duration atributo seja necessário para cada EXT-X-CUE-OUT tag.	5 de agosto de 2020
Parâmetro do modo de relatório de inicialização da sessão	Foi adicionado suporte para o reportingMode parâmetro nas solicitações de inicialização da sessão. Agora você pode especificar se MediaTailor realiza o rastreamento de anúncios do lado do cliente ou do lado do servidor ao inicializar uma sessão.	30 de julho de 2020
Novo tópico sobre pára-choques	Adicionado um novo tópico sobre pára-choques	27 de julho de 2020
A supressão de anúncios está disponível para DASH	A supressão de anúncios agora está disponível para o DASH. A restrição "HLS-only" foi removida do tópico de supressão de anúncios.	3 de junho de 2020
Atualizar nomes específicos do console	Os nomes específicos ao console foram atualizados para refletir uma versão mais recente da interface do usuário do console.	1º de maio de 2020
Nova variável de anúncio avail.index dinâmico	Adição de suporte para a nova variável de anúncio dinâmicos de dados de sessão avail.index .	13 de março de 2020

Novos AdVerifications e Extensions elementos	Para relatórios do lado do cliente, os elementos AdVerifications e Extensions são compatíveis.	10 de março de 2020
Configuração do limite de personalização	Adicionamos suporte para a configuração opcional de limite de personalização.	14 de fevereiro de 2020
Manifestos DASH VOD	Suporte adicionado para manifestos DASH de vídeo sob demanda (VOD) do servidor de origem, com saída de manifesto de vários períodos.	23 de dezembro de 2019
Suporte de console para transcodificar o nome do perfil	Adicionada descrição para o nome do perfil de transcodificação na configuração.	23 de dezembro de 2019
Tabelas de limites atualizadas	Atualização dos limites para redirecionamentos do ADS e tempos limite do ADS.	18 de dezembro de 2019
Documento comportamentos de pré-lançamento ao vivo	Foi adicionada a seção de inserção de Pre-Roll anúncios para descrever como os anúncios pré-lançados ao vivo funcionam com. AWS Elemental MediaTailor	26 de novembro de 2019
Support para anúncios pré-lançados ao vivo	Adição de suporte para inserção de anúncios pre-roll no início de um streaming ao vivo.	11 de setembro de 2019

Análise de registros do ADS no Amazon CloudWatch Logs Insights	Foram adicionadas informações para usar os registros do AWS Elemental MediaTailor ADS e o CloudWatch Logs Insights para analisar suas MediaTailor sessões.	13 de agosto de 2019
Novo capítulo de segurança	Adição de um capítulo para aprimorar a segurança e padronizar a cobertura.	23 de maio de 2019
Manifestos de período único do DASH	Suporte adicionado para manifestos DASH de um único período no servidor de origem, com saída de manifesto de vários períodos.	4 de abril de 2019
Support para SCTE-35 UPIDs no URL do ADS	Adição de suporte para inclusão de um ID exclusivo de programa (UPID) no URL do Ad Decision Server (ADS – Servidor de decisões de anúncios). Isso permite que o ADS forneça direcionamento de anúncios em nível de programa em um streaming linear ao vivo.	28 de março de 2019
Client-side os relatórios oferecem suporte a anúncios complementares	Para relatórios do lado do cliente, a resposta do URL AWS Elemental MediaTailor de rastreamento agora inclui metadados de anúncios complementares.	28 de março de 2019

Documentação sobre HLS e marcadores	Adicionada uma seção que descreve os marcadores de anúncios HLS com suporte.	1 de março de 2019
Suporte a marcação	Adição de suporte para tags de recursos na configuração no AWS Elemental MediaTailor. A marcação permite que você identifique e organize seus AWS recursos, controle o acesso a eles e acompanhe seus AWS custos.	14 de fevereiro de 2019
Informações de AWS CloudTrail registro adicionadas	Foi adicionado um tópico sobre CloudTrail como usar para registrar ações na AWS Elemental MediaTailor API.	11 de fevereiro de 2019
Seção adicionada sobre erros de reprodução	Adição de Informações sobre os erros que podem ser retornados pelo MediaTailor durante a reprodução, em resposta a solicitações de um player ou uma rede de entrega de conteúdo (CDN).	4 de fevereiro de 2019
Binário codificado em base64 do DASH	Adicionado suporte para fornecer informações por emenda em manifestos em binário codificado em base64, em marcadores <scte35:Signal> <scte35:Binary> .	4 de janeiro de 2019

Sinal de horário DASH	Adição de suporte para fornecer informações por emenda em manifestos em marcadores <code><scte35:TimeSignal></code> .	5 de dezembro de 2018
Suporte de localização do DASH	Foi adicionado suporte para a MPEG-DASH <code><Location></code> tag.	4 de dezembro de 2018
Suporte ao DASH	Foi adicionado suporte para MPEG-DASH manifestos.	14 de novembro de 2018
Tabelas de limites atualizadas	Atualização de limites para configurações e tamanho do manifesto.	13 de outubro de 2018
Métricas novas e atualizadas	Inclusão de métricas do Ad Decision Server (ADS, Servidor de decisões de anúncios) e tempos limite de origem e atualização das definições de erros para incluir respostas expiradas.	13 de outubro de 2018
Melhor cobertura de documentação para casos de uso de inserção de anúncios do lado do servidor e do lado do cliente	Descrição expandida e exemplos para abordar o uso de variáveis de anúncios dinâmicos para a inserção de anúncios no lado do servidor e a inserção no lado do cliente.	1º de outubro de 2018
Novas regiões	Adição de suporte para as regiões PDX e FRA.	18 de julho de 2018
VAST/VPAID	Adição de informações sobre VAST e VPAID.	16 de março de 2018

CloudWatch	Foram adicionadas informações sobre CloudWatch métricas, namespaces e dimensões disponíveis.	16 de março de 2018
Novas regiões	Adição de suporte para as regiões Ásia-Pacífico (Cingapura), Ásia-Pacífico (Cingapura) e Ásia-Pacífico (Tóquio).	8 de fevereiro de 2018
Informações sobre a política do IAM	Foram adicionadas informações específicas da política do IAM para AWS Elemental MediaTailor. Instruções adicionadas para criar funções não administrativas com permissões limitadas.	3 de janeiro de 2018
Primeiro lançamento	Primeira versão desta documentação.	27 de novembro de 2017

Note

- Os Serviços de AWS Mídia não foram projetados ou destinados ao uso com aplicativos ou em situações que exijam desempenho à prova de falhas, como operações de segurança de vida, sistemas de navegação ou comunicação, controle de tráfego aéreo ou máquinas de suporte de vida nas quais a indisponibilidade, interrupção ou falha dos serviços possa levar à morte, ferimentos pessoais, danos materiais ou danos ambientais.

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.