



Guia do Usuário

AWS Elemental MediaStore



AWS Elemental MediaStore: Guia do Usuário

Copyright © 2023 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

As marcas comerciais e imagens comerciais 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 clientes ou que deprecie ou desprestigie 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 o MediaStore?	1
Conceitos e terminologia	1
Serviços relacionados	3
Acesso ao MediaStore	3
Preços	4
Regiões e endpoints	4
Configuração do AWS Elemental MediaStore	6
Inscrever-se para uma Conta da AWS	6
Crie um usuário administrador	7
Conceitos básicos	8
Etapa 1: Acesso ao AWS Elemental MediaStore	8
Etapa 2: Criar um contêiner	8
Etapa 3: Fazer upload de um objeto	9
Etapa 4: Acessar um objeto	10
Contêineres	11
Regras para nomes de contêineres	11
Criar um contêiner	11
Visualizar detalhes do contêiner	13
Visualizar uma lista de contêineres	14
Excluir um contêiner	15
Políticas	16
Políticas de contêiner	16
Visualizar uma política de contêiner	17
Editar uma política de contêiner	18
Exemplos de políticas de contêiner	19
Políticas de CORS	25
Cenários de casos de uso	26
Adicionar uma política de CORS	27
Visualizar uma política de CORS	28
Editar uma política de CORS	29
Excluir uma política de CORS	30
Solução de problemas	31
Exemplos de políticas de CORS	32
Políticas de ciclo de vida de objetos	33

Componentes de uma política de ciclo de vida de objetos	34
Adicionar uma política de ciclo de vida de objetos	41
Visualizar uma política de ciclo de vida de objetos	42
Editar uma política de ciclo de vida de objetos	44
Excluir uma política de ciclo de vida de objetos	45
Exemplos de política de ciclo de vida de objetos	45
Políticas de métrica	50
Adicionar uma política de métrica	51
Visualizar uma política de métrica	51
Editar uma política de métrica	51
Exemplos de políticas de métrica	52
Pastas	56
Regras para nomes de pastas	56
Criar uma pasta	57
Excluir uma pasta	57
Objetos	58
Fazer upload de um objeto	58
Visualizar uma lista	60
Visualizar detalhes do objeto	63
Fazer download de um objeto	64
Excluir objetos	65
Excluir um objeto	65
Esvaziar um contêiner	66
Segurança	68
Proteção de dados	69
Criptografia de dados	70
Identity and Access Management	70
Público	71
Autenticando com identidades	71
Como gerenciar acesso usando políticas	75
Como o AWS Elemental MediaStore funciona com o IAM	78
Exemplos de políticas baseadas em identidade	86
Solução de problemas	89
Registrar em log e monitoramento	91
Alarmes do Amazon CloudWatch	91
Logs do AWS CloudTrail	91

AWS Trusted Advisor	92
Validação de compatibilidade	92
Resilience	93
Infrastructure Security	94
Prevenção contra o ataque do “substituto confuso” em todos os serviços	94
Monitorar e atribuir tags (tagging)	96
Registrar em log chamadas de API com o CloudTrail	97
Informações sobre o MediaStore no CloudTrail	97
Exemplo: entradas de arquivo de log	99
Monitoramento com CloudWatch	100
CloudWatch Logs	101
CloudWatch Events	111
Métricas do CloudWatch	115
Marcação	119
Recursos compatíveis no AWS Elemental MediaStore	120
Convenções de uso e nomenclatura de tags	120
Gerenciar tags	121
Como trabalhar com CDNs	122
Permitir que o CloudFront acesse seu contêiner	122
Uso do controle de acesso de origem (OAC)	123
Uso de segredos compartilhados	123
Interação do MediaStore com caches HTTP	126
Solicitações condicionais	127
Cotas	128
Informações relacionadas	131
Histórico do documento	132
Glossário do AWS	137

O que é o AWS Elemental MediaStore?

O AWS Elemental MediaStore é um serviço de originação e armazenamento de vídeo que oferece alto desempenho e consistência imediata exigida para originação ao vivo. Com o MediaStore, você pode gerenciar ativos de vídeo como objetos em contêineres para criar fluxos de trabalho de mídia confiáveis baseados em nuvem.

Para usar o serviço, faça upload de seus objetos a partir de uma origem, como um codificador ou feed de dados, para um contêiner criado no MediaStore.

O MediaStore é uma ótima opção para armazenar arquivos de vídeo fragmentados quando você precisa de consistência forte, leituras e gravações de baixa latência e capacidade de lidar com grandes volumes de solicitações simultâneas. Se você não entrega vídeos de streaming ao vivo, considere usar o [Amazon Simple Storage Service \(Amazon S3\)](#) em vez disso.

Tópicos

- [Conceitos e terminologia do AWS Elemental MediaStore](#)
- [Serviços relacionados](#)
- [Acesso ao AWS Elemental MediaStore](#)
- [Definição de preço do AWS Elemental MediaStore](#)
- [Regiões e endpoints do AWS Elemental MediaStore](#)

Conceitos e terminologia do AWS Elemental MediaStore

ARN

Um [Nome de recurso da Amazon](#).

Corpo

Os dados a serem carregados em um objeto.

Intervalo (bytes)

Um subconjunto de dados do objeto a ser endereçado. Para obter mais informações, consulte [intervalo](#) da especificação HTTP.

Contêiner

Um namespace que contém objetos. Um contêiner possui um endpoint que você pode usar para gravar e recuperar objetos, além de anexar políticas de acesso.

Endpoint

Um ponto de entrada para o serviço do MediaStore, fornecido como um URL raiz de HTTPS.

ETag

Uma [tag de entidade](#), que é um hash de dados de objeto.

Pasta

Uma divisão de um contêiner. Uma pasta pode conter objetos e outras pastas.

Item

Um termo usado para se referir a objetos e pastas.

Objeto

Um ativo, semelhante a um [objeto do Amazon S3](#). Os objetos são as entidades fundamentais armazenadas no MediaStore. O serviço aceita todos os tipos de arquivo.

Serviço de origem

O MediaStore é considerado um serviço de origem porque é o ponto de distribuição para entrega de conteúdo de mídia.

Path

Um identificador exclusivo para um objeto ou pasta, que indica sua localização no contêiner.

Peça

Um subconjunto de dados (bloco) de um objeto.

Política

Uma [política do IAM](#).

Recurso

Uma entidade na AWS com a qual você pode trabalhar. Cada recurso da AWS recebe um nome de recurso da Amazon (ARN) que atua como um identificador exclusivo. No MediaStore, esse é o recurso e o formato de ARN:

- Contêiner: `aws:mediastore:region:account-id:container/:containerName`

Serviços relacionados

- O Amazon CloudFront é um serviço de rede de entrega de conteúdo (CDN) global que entrega dados e vídeos com segurança aos visualizadores. Use o CloudFront para entregar conteúdo com o melhor desempenho possível. Para obter mais informações, consulte o [Guia do desenvolvedor do Amazon CloudFront](#).
- O AWS CloudFormation é um serviço que ajuda você a modelar e configurar seus recursos da AWS. Crie um modelo que descreve todos os recursos do AWS que você deseja (como contêineres do MediaStore) e o AWS CloudFormation cuida do provisionamento e da configuração desses recursos para você. Não é necessário criar e configurar individualmente os recursos da AWS e descobrir o que depende do que; o AWS CloudFormation lida com tudo isso. Para obter mais informações, consulte o [Guia do usuário do AWS CloudFormation](#).
- O AWS CloudTrail é um serviço que permite monitorar as chamadas feitas para a API do CloudTrail de sua conta, incluindo chamadas feitas pelo Console de Gerenciamento da AWS, pela AWS CLI e outros serviços. Para obter mais informações, consulte o [Guia do usuário do AWS CloudTrail](#).
- O Amazon CloudWatch é um serviço de monitoramento para recursos da Nuvem AWS e dos aplicativos que você executa na AWS. Use CloudWatch Events para monitorar alterações no status de contêineres e de objetos no MediaStore. Para obter mais informações, consulte a [documentação do Amazon CloudWatch](#).
- O AWS Identity and Access Management (IAM) é um serviço da Web que ajuda a controlar seguramente o acesso de seus usuários aos recursos da AWS. Use o IAM para controlar quem pode usar os recursos da AWS (autenticação) e quais recursos os usuários podem usar de quais maneiras (autorização). Para obter mais informações, consulte [Configuração do AWS Elemental MediaStore](#).
- O Amazon Simple Storage Service (Amazon S3) é um armazenamento de objeto feito para armazenar e recuperar qualquer quantidade de dados de qualquer lugar. Para mais informações, consulte a [documentação do Amazon S3](#).

Acesso ao AWS Elemental MediaStore

Você pode acessar o MediaStore usando qualquer um dos seguintes métodos:

- **AWS Management Console:** os procedimentos descritos neste guia explicam como usar o Console de Gerenciamento da AWS a fim de executar tarefas para o MediaStore. Para acessar o MediaStore usando o console:

```
https://<region>.console.aws.amazon.com/mediastore/home
```

- **AWS Command Line Interface:** para obter mais informações, consulte o [Guia do usuário do AWS Command Line Interface](#). Para acessar o MediaStore usando o endpoint da CLI:

```
aws mediastore
```

- **API do MediaStore:** se você estiver usando uma linguagem de programação para a qual um SDK não está disponível, consulte a [Referência da API AWS Elemental MediaStore](#) para obter informações sobre as ações de API e sobre como fazer solicitações de API. Para acessar o MediaStore usando o endpoint da API REST:

```
https://mediastore.<region>.amazonaws.com
```

- **SDKs da AWS:** se estiver usando uma linguagem de programação para a qual a AWS fornece um SDK, você poderá usar um SDK para acessar o MediaStore. Os SDKs simplificam a autenticação, integram-se com facilidade ao ambiente de desenvolvimento e fornecem acesso simples aos comandos do MediaStore. Para obter mais informações, consulte [Ferramentas para a Amazon Web Services](#).
- **Ferramentas da AWS para Windows PowerShell:** para obter mais informações, consulte o [Guia do Usuário do AWS Tools for Windows PowerShell](#).

Definição de preço do AWS Elemental MediaStore

Assim como ocorre com outros produtos da AWS, não há contratos nem compromissos mínimos para uso do MediaStore. É cobrada uma taxa por GB consumidos quando o conteúdo entra no serviço e uma taxa mensal por GB para o conteúdo armazenado no serviço. Para mais informações, consulte [Precificação do AWS Elemental MediaStore](#).

Regiões e endpoints do AWS Elemental MediaStore

Para reduzir a latência de dados nos aplicativos, o MediaStore oferece um endpoint regional para você fazer sua solicitação:

```
https://mediastore.<region>.amazonaws.com
```

Para ver a lista completa de regiões da AWS em que o MediaStore está disponível, consulte [Endpoints e cotas do AWS Elemental MediaStore](#) na Referência geral da AWS.

Configuração do AWS Elemental MediaStore

Esta seção orienta você sobre as etapas necessárias para configurar os usuários que terão acesso ao AWS Elemental MediaStore. Para informações detalhadas e adicionais sobre o Identity and Access Management (IAM) para o MediaStore, consulte [Identity and Access Management para o AWS Elemental MediaStore](#).

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

Tópicos

- [Inscrever-se para uma Conta da AWS](#)
- [Crie um usuário administrador](#)

Inscrever-se para uma Conta da AWS

Se você ainda não tem uma Conta da AWS, siga as etapas a seguir para criar uma.

Para se cadastrar em uma Conta da AWS

1. Abra <https://portal.aws.amazon.com/billing/signup>.
2. Siga as instruções on-line.

Parte do procedimento de inscrição envolve receber uma chamada telefônica e digitar um código de verificação no teclado do telefone.

Quando você se cadastra em uma Conta da AWS, um Usuário raiz da conta da AWS é criado. O usuário raiz tem acesso a todos os Serviços da AWS e recursos na conta. Como prática recomendada de segurança, [atribua acesso administrativo a um usuário administrativo](#) e use somente o usuário raiz para realizar as [tarefas que exigem acesso do usuário raiz](#).

A AWS envia um e-mail de confirmação depois que o processo de cadastramento é concluído. A qualquer momento, é possível visualizar as atividades da conta atual e gerenciar sua conta acessando <https://aws.amazon.com/> e selecionando Minha conta.

Crie um usuário administrador

Depois de se inscrever em uma Conta da AWS, proteja o Usuário raiz da conta da AWS, ative o AWS IAM Identity Center e crie um usuário administrativo para não usar o usuário raiz em tarefas cotidianas.

Proteger seu Usuário raiz da conta da AWS

1. Faça login no [AWS Management Console](#) como o proprietário da conta ao escolher a opção Usuário raiz e inserir o endereço de e-mail da Conta da AWS. Na próxima página, digite sua senha.

Para obter ajuda ao fazer login usando o usuário raiz, consulte [Fazer login como usuário raiz](#) no Guia do usuário do Início de Sessão da AWS.

2. Ative a autenticação multifator (MFA) para o usuário raiz.c

Para obter instruções, consulte [Habilitar um dispositivo MFA virtual para o usuário raiz de sua conta da Conta da AWS para seu \(console\)](#) no Guia do usuário do IAM.

Criar um usuário administrativo

1. Habilitar o IAM Identity Center.

Para obter instruções, consulte [Ativar AWS IAM Identity Center](#) no Guia do usuário do AWS IAM Identity Center.

2. No Centro de Identidade do IAM, conceda acesso administrativo a um usuário administrativo.

Para ver um tutorial sobre como usar o Diretório do Centro de Identidade do IAM como fonte de identidade, consulte [Configurar o acesso do usuário com o Diretório do Centro de Identidade do IAM padrão](#) no Guia do usuário do AWS IAM Identity Center.

Login como usuário administrativo

- Para fazer login com seu usuário do Centro de Identidade do IAM, use a URL de login que foi enviada ao seu endereço de e-mail quando você criou o usuário do Centro do Usuário do IAM.

Para obter ajuda com o login utilizando um usuário do Centro de Identidade do IAM, consulte [Fazer login no portal de acesso da AWS](#), no Guia do usuário do Início de Sessão da AWS.

Conceitos básicos do AWS Elemental MediaStore

Este tutorial de conceitos básicos mostra como usar o AWS Elemental MediaStore para criar um contêiner e fazer upload de um objeto.

Tópicos

- [Etapa 1: Acesso ao AWS Elemental MediaStore](#)
- [Etapa 2: Criar um contêiner](#)
- [Etapa 3: Fazer upload de um objeto](#)
- [Etapa 4: Acessar um objeto](#)

Etapa 1: Acesso ao AWS Elemental MediaStore

Depois de configurar sua conta AWS e criar usuários e funções, faça login no console do AWS Elemental MediaStore.

Para acessar o AWS Elemental MediaStore

- Entre no AWS Management Console e abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.

Note

Você pode fazer login usando qualquer uma das credenciais do IAM que você criou para essa conta. Para obter informações sobre como criar credenciais do IAM, consulte [Configuração do AWS Elemental MediaStore](#).

Etapa 2: Criar um contêiner

Use contêineres no AWS Elemental MediaStore para armazenar suas pastas e objetos. Você pode usar contêineres para agrupar objetos relacionados, da mesma forma que usa um diretório para agrupar arquivos em um sistema de arquivos. Você não é cobrado quando cria contêineres, apenas quando faz upload de um objeto para um contêiner.

Para criar um contêiner

1. Na página Containers (Contêineres), selecione Create container (Criar contêiner).
2. Em Container name (nome do contêiner), digite um nome para o contêiner. Para obter mais informações, consulte [Regras para nomes de contêineres](#).
3. Escolha Criar contêiner. O AWS Elemental MediaStore adiciona o novo contêiner a uma lista de contêineres. Inicialmente, o status do contêiner é Creating (Criando) e, depois, ele muda para Active (Ativo).

Etapa 3: Fazer upload de um objeto

Você pode fazer upload de objetos (até 25 MB cada) para um contêiner ou para uma pasta em um contêiner. Para fazer upload de um objeto para uma pasta, você especifica o caminho para a pasta. Se a pasta já existir, o AWS Elemental MediaStore armazenará o objeto na pasta. Se a pasta não existir, o serviço a cria e, em seguida, armazena o objeto na pasta.

Note

Os nomes de arquivo de objeto podem conter apenas letras, números, pontos (.), sublinhados (_), tils (~) e hifens (-).

Para carregar um objeto

1. Na página Containers (Contêineres), selecione o nome do contêiner que você acabou de criar. A página de detalhes do contêiner é exibida.
2. Selecione Upload object (Fazer upload de objeto).
3. Em Target path (Caminho de destino), digite o caminho das pastas. Por exemplo, premium/canada. Se alguma das pastas no caminho ainda não existir, o AWS Elemental MediaStore as criará automaticamente.
4. Em Object (Objeto), selecione Browse (Navegar).
5. Navegue até a pasta apropriada e escolha um objeto para fazer upload.
6. Selecione Open (Abrir) e Upload (Fazer upload).

Etapa 4: Acessar um objeto

Você pode fazer download de seus objetos para um endpoint especificado.

1. Na página Containers (Contêineres), selecione o nome do contêiner que tem o objeto do qual você deseja fazer download.
2. Se o objeto do qual você deseja fazer download estiver em uma subpasta, continue escolhendo os nomes das pastas até ver o objeto.
3. Escolha o nome do objeto.
4. Na página de detalhes do objeto, selecione Download (Fazer download).

Contêineres no AWS Elemental MediaStore

Você usa contêineres no MediaStore para armazenar as pastas e os objetos. Objetos relacionados podem ser agrupados em contêineres da mesma forma que você usa um diretório para agrupar arquivos em um sistema de arquivos. Você não é cobrado quando cria contêineres, apenas quando faz upload de um objeto para um contêiner. Para obter mais informações sobre cobranças, consulte [Preços do AWS Elemental MediaStore](#).

Tópicos

- [Regras para nomes de contêineres](#)
- [Criar um contêiner](#)
- [Visualizar os detalhes de um contêiner](#)
- [Visualizar uma lista de contêineres](#)
- [Excluir um contêiner](#)

Regras para nomes de contêineres

Ao escolher um nome para o contêiner, lembre-se do seguinte:

- O nome deve ser exclusivo dentro da conta atual para a região da AWS atual.
- O nome pode conter letras maiúsculas, minúsculas, números e sublinhados (_).
- O nome deve ter de 1 a 255 caracteres.
- Os nomes diferenciam letras maiúsculas de minúsculas. Por exemplo, você pode ter um contêiner chamado `myContainer` e uma pasta chamada `mycontainer` porque esses nomes são exclusivos.
- Um contêiner não pode ser renomeados depois de criado.

Criar um contêiner

Você pode criar até 100 contêineres para cada conta da AWS. É possível criar quantas pastas quiser, desde que não estejam aninhadas mais de 10 níveis em um contêiner. Além disso, você pode fazer upload de quantos objetos quiser para cada contêiner.

Tip

Você também pode criar um contêiner automaticamente usando um modelo do AWS CloudFormation. O modelo do AWS CloudFormation gerencia os dados para cinco ações de API: criação de um contêiner, definição do registro de acesso, atualização da política de contêiner padrão, adição de uma política de CORS (compartilhamento de recursos entre origens) e adição de uma política de ciclo de vida de objetos. Para obter mais informações, consulte o [Guia do usuário do AWS CloudFormation](#).

Para criar um contêiner (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione Create container (Criar contêiner).
3. Em nome do Container (Contêiner), insira um nome para o contêiner. Para obter mais informações, consulte [Regras para nomes de contêineres](#).
4. Selecione Criar contêiner. O AWS Elemental MediaStore adiciona o novo contêiner a uma lista de contêineres. Inicialmente, o status do contêiner é Creating (Criando) e, depois, ele muda para Active (Ativo).

Para criar um contêiner (AWS CLI)

- Na AWS CLI, use o comando `create-container`:

```
aws mediastore create-container --container-name ExampleContainer --region us-west-2
```

O exemplo a seguir mostra o valor de retorno:

```
{
  "Container": {
    "AccessLoggingEnabled": false,
    "CreationTime": 1563557265.0,
    "Name": "ExampleContainer",
    "Status": "CREATING",
    "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer"
  }
}
```

```
}
```

Visualizar os detalhes de um contêiner

Os detalhes de um contêiner incluem a política, o endpoint, o ARN e a data de criação do contêiner.

Para visualizar os detalhes de um contêiner (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner.

A página de detalhes do contêiner é exibida. Esta página é dividida em duas seções:

- A seção Objects (Objetos), que lista os objetos e as pastas no contêiner.
- A seção de política do Container (Contêiner), que mostra a política baseada em recursos associada a esse contêiner. Para obter informações sobre políticas de recursos, consulte [Políticas de contêiner](#).

Para visualizar os detalhes de um contêiner (AWS CLI)

- Na AWS CLI, use o comando `describe-container`:

```
aws mediastore describe-container --container-name ExampleContainer --region us-west-2
```

O exemplo a seguir mostra o valor de retorno:

```
{
  "Container": {
    "CreationTime": 1563558086.0,
    "AccessLoggingEnabled": false,
    "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer",
    "Status": "ACTIVE",
    "Name": "ExampleContainer",
    "Endpoint": "https://aaabbbcccddee.data.mediastore.us-
west-2.amazonaws.com"
  }
}
```

```
}
```

Visualizar uma lista de contêineres

Você pode visualizar uma lista de todos os contêineres que estão associados à sua conta.

Para visualizar uma lista de contêineres (console)

- Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.

A página Containers (Contêineres) é exibida, listando todos os contêineres associados à sua conta.

Para visualizar uma lista de contêineres (AWS CLI)

- Na AWS CLI, use o comando `list-containers`.

```
aws mediastore list-containers --region us-west-2
```

O exemplo a seguir mostra o valor de retorno:

```
{
  "Containers": [
    {
      "CreationTime": 1505317931.0,
      "Endpoint": "https://aaabbbcccddee.data.mediastore.us-
west-2.amazonaws.com",
      "Status": "ACTIVE",
      "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleLiveDemo",
      "AccessLoggingEnabled": false,
      "Name": "ExampleLiveDemo"
    },
    {
      "CreationTime": 1506528818.0,
      "Endpoint": "https://fffggghhhiiijj.data.mediastore.us-
west-2.amazonaws.com",
      "Status": "ACTIVE",
      "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer",

```

```
        "AccessLoggingEnabled": false,  
        "Name": "ExampleContainer"  
    }  
]  
}
```

Excluir um contêiner

Você só pode excluir um contêiner se ele não tiver objetos.

Para excluir um contêiner (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione a opção à esquerda do nome do contêiner.
3. Escolha Delete (Excluir).

Para excluir um contêiner (AWS CLI)

- Na AWS CLI, use o comando `delete-container`:

```
aws mediastore delete-container --container-name=ExampleLiveDemo --region us-west-2
```

Este comando não possui valor de retorno.

Políticas no AWS Elemental MediaStore

É possível aplicar uma ou mais dessas políticas ao seu contêiner do AWS Elemental MediaStore:

- [Política de contêiner](#): define direitos de acesso a todas as pastas e objetos dentro do contêiner. O MediaStore define uma política padrão que permite aos usuários realizarem todas as operações do MediaStore no contêiner. Essa política especifica que todas as operações devem ser executadas por HTTPS. Depois de criar um contêiner, você poderá editar a política de contêiner.
- [Política de compartilhamento entre origens \(CORS\)](#): permite que as aplicações web do cliente de um domínio interajam com recursos em outro domínio. O MediaStore não define uma política CORS padrão.
- [Política de métricas](#): permite que o MediaStore envie métricas para o Amazon CloudWatch. O MediaStore não define uma política de métrica padrão.
- [Política de ciclo de vida de objetos](#): controla por quanto tempo os objetos permanecem em um contêiner do MediaStore. O MediaStore não define uma política padrão de ciclo de vida do objeto.

Políticas de contêiner no AWS Elemental MediaStore

Cada contêiner tem uma política baseada em recursos que controla os direitos de acesso a todas as pastas e objetos desse contêiner. A política padrão, que é automaticamente anexada a todos os novos contêineres, permite acesso a todas as operações do AWS Elemental MediaStore no contêiner. Ela especifica que esse acesso tem a condição de exigir HTTPS para as operações. Depois de criar um contêiner, você pode editar a política anexada a ele.

Também é possível especificar uma [política de ciclo de vida de objetos](#) que controla a data de expiração de objetos em um contêiner. Depois de os objetos chegarem à idade máxima especificada, o serviço excluirá os objetos do contêiner.

Tópicos

- [Visualizar uma política de contêiner](#)
- [Editar uma política de contêiner](#)
- [Exemplos de políticas de contêiner](#)

Visualizar uma política de contêiner

Você pode usar o console ou a AWS CLI para visualizar a política baseada em recursos de um contêiner.

Para visualizar uma política de contêiner (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), escolha o nome do contêiner.

A página de detalhes do contêiner é exibida. A política é exibida na seção Container policy (Política de contêiner).

Para visualizar uma política de contêiner (AWS CLI)

- Na AWS CLI, use o comando `get-container-policy`:

```
aws mediastore get-container-policy --container-name ExampleLiveDemo --region us-west-2
```

O exemplo a seguir mostra o valor de retorno:

```
{
  "Policy": {
    "Version": "2012-10-17",
    "Statement": [
      {
        "Sid": "PublicReadOverHttps",
        "Effect": "Allow",
        "Principal": {
          "AWS": "arn:aws:iam::111122223333:root",
        },
        "Action": [
          "mediastore:GetObject",
          "mediastore:DescribeObject",
        ],
        "Resource": "arn:aws:mediastore:us-west-2:111122223333:container/ExampleLiveDemo/*",
        "Condition": {
          "Bool": {
            "aws:SecureTransport": "true"
          }
        }
      }
    ]
  }
}
```

```
    }
  }
}
]
```

Editar uma política de contêiner

Você pode editar as permissões na política de contêiner padrão ou pode criar uma nova política que substitui a política padrão. Leva até cinco minutos para que a nova política tenha efeito.

Para editar uma política de contêiner (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), escolha o nome do contêiner.
3. Escolha Editar política. Para obter exemplos que mostram como definir permissões diferentes, consulte [the section called “Exemplos de políticas de contêiner”](#).
4. Faça as alterações apropriadas e, em seguida, selecione Save (Salvar).

Para editar uma política de contêiner (AWS CLI)

1. Crie um arquivo que defina a política de contêiner:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadOverHttps",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal": "*",
      "Resource": "arn:aws:mediastore:us-  
west-2:111122223333:container/ExampleLiveDemo/*",
      "Condition": {
        "Bool": {
          "aws:SecureTransport": "true"
        }
      }
    }
  ]
}
```

```
]
}
```

2. Na AWS CLI, use o comando `put-container-policy`:

```
aws mediastore put-container-policy --container-name ExampleLiveDemo --
policy file://ExampleContainerPolicy.json --region us-west-2
```

Este comando não possui valor de retorno.

Exemplos de políticas de contêiner

Os exemplos a seguir mostram políticas de contêiner construídas para diferentes grupos de usuários.

Tópicos

- [Exemplo de política de contêiner: padrão](#)
- [Exemplo de política de contêiner: acesso público de leitura por HTTPS](#)
- [Exemplo de política de contêiner: acesso público de leitura por HTTP ou HTTPS](#)
- [Exemplo de política de contêiner: acesso de leitura entre contas – HTTP habilitado](#)
- [Exemplo de política de contêiner: acesso de leitura entre contas por HTTPS](#)
- [Exemplo de política de contêiner: acesso de leitura entre contas a uma função](#)
- [Exemplo de política de contêiner: acesso total entre contas a uma função](#)
- [Exemplo de política de contêiner: acesso restrito a endereços IP específicos](#)

Exemplo de política de contêiner: padrão

Quando você cria um contêiner, o AWS Elemental MediaStore anexa automaticamente a seguinte política baseada em recursos:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "MediaStoreFullAccess",
      "Action": [ "mediastore:*" ],
      "Principal": {
        "AWS" : "arn:aws:iam::<aws_account_number>:root"},
    }
  ]
}
```



```

    "Effect": "Allow",
    "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
name>/*",
    "Condition": {
      "Bool": { "aws:SecureTransport": "true" }
    }
  }
]
}

```

A política é criada no serviço, de modo que você não precisa criá-la. Porém, você pode [editar a política](#) no contêiner se as permissões na política padrão não estiverem alinhadas com as permissões que você deseja usar para o contêiner.

A política padrão que é atribuída a todos os novos contêineres permite acesso a todas as operações do MediaStore no contêiner. Ela especifica que esse acesso tem a condição de exigir HTTPS para as operações.

Exemplo de política de contêiner: acesso público de leitura por HTTPS

Essa política de exemplo permite que os usuários recuperem um objeto por meio de uma solicitação HTTPS. Ela permite acesso de leitura a qualquer pessoa por meio de uma conexão segura SSL/TLS: usuários autenticados e usuários anônimos (usuários que não estiverem conectados). A instrução tem o nome `PublicReadOverHttps`. Permite acesso às operações `GetObject` e `DescribeObject` em qualquer objeto, conforme especificado pelo `*` no final do caminho do recurso. Ela especifica que esse acesso tem a condição de exigir HTTPS para as operações:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadOverHttps",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal": "*",
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
name>/*",
      "Condition": {
        "Bool": {
          "aws:SecureTransport": "true"
        }
      }
    }
  ]
}

```

```

    }
  ]
}

```

Exemplo de política de contêiner: acesso público de leitura por HTTP ou HTTPS

Essa política de exemplo permite acesso às operações `GetObject` e `DescribeObject` em qualquer objeto (conforme especificado pelo `*` no final do caminho do recurso). Ela permite acesso de leitura a qualquer pessoa, incluindo todos os usuários autenticados e usuários anônimos (usuários que não estão conectados):

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadOverHttpOrHttps",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal": "*",
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>;container/<container name>/*",
      "Condition": {
        "Bool": { "aws:SecureTransport": ["true", "false"] }
      }
    }
  ]
}

```

Exemplo de política de contêiner: acesso de leitura entre contas – HTTP habilitado

Essa política de exemplo permite que os usuários recuperem um objeto por meio de uma solicitação HTTP. Ela permite esse acesso a usuários autenticados com acesso entre contas. O objeto não precisa estar hospedado em um servidor com um certificado SSL/TLS:

```

{
  "Version" : "2012-10-17",
  "Statement" : [ {
    "Sid" : "CrossAccountReadOverHttpOrHttps",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::<other acct number>;root"
    },

```

```

    "Action" : [ "mediastore:GetObject", "mediastore:DescribeObject" ],
    "Resource" : "arn:aws:mediastore:<region>:<owner acct number>:container/<container
name>/*",
    "Condition" : {
      "Bool" : {
        "aws:SecureTransport" : [ "true", "false" ]
      }
    }
  } ]
}

```

Exemplo de política de contêiner: acesso de leitura entre contas por HTTPS

Este exemplo de política permite acessar as operações `GetObject` e `DescribeObject` em qualquer objeto (conforme especificado pelo `*` no final do caminho do recurso) que pertence ao usuário raiz do <outro número de conta> especificado. Ela especifica que esse acesso tem a condição de exigir HTTPS para as operações:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CrossAccountReadOverHttps",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal":{
        "AWS": "arn:aws:iam::<other acct number>:root"},
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container
name>/*",
      "Condition": {
        "Bool": {
          "aws:SecureTransport": "true"
        }
      }
    }
  ]
}

```

Exemplo de política de contêiner: acesso de leitura entre contas a uma função

O exemplo de política permite acessar as operações `GetObject` e `DescribeObject` em qualquer objeto (conforme especificado pelo `*` no final do caminho do recurso) que pertence ao <número de

conta do proprietário>. Ela permite esse acesso a qualquer usuário do <outro número de conta> se essa conta tiver assumido a função especificada no <nome da função>:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CrossAccountRoleRead",
      "Effect": "Allow",
      "Action": ["mediastore:GetObject", "mediastore:DescribeObject"],
      "Principal": {
        "AWS": "arn:aws:iam::<other acct number>:role/<role name>"
      },
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container name>/*",
    }
  ]
}
```

Exemplo de política de contêiner: acesso total entre contas a uma função

Essa política de exemplo permite acesso entre contas para atualizar qualquer objeto na conta, desde que o usuário esteja conectado por HTTP. Ela também permite acesso entre contas para excluir, fazer download e descrever os objetos por HTTP ou HTTPS em uma conta que tenha assumido a função especificada:

- A primeira instrução é `CrossAccountRolePostOverHttps`. Ela permite o acesso à operação `PutObject` em qualquer objeto e permite esse acesso a qualquer usuário da conta especificada, se essa conta tiver assumido a função especificada em <nome da função>. Ela especifica que esse acesso tem a condição de exigir HTTPS para a operação, essa condição deve sempre ser incluída ao fornecer acesso à operação `PutObject`.

Em outras palavras, qualquer principal que tenha acesso a contas cruzadas pode acessar a operação `PutObject`, mas somente por HTTPS.

- A segunda instrução é `CrossAccountFullAccessExceptPost`. Permite acesso a todas as operações, exceto `PutObject` em qualquer objeto. Permite esse acesso a qualquer usuário da conta especificada se essa conta tiver assumido a função especificada no <nome da função>. Esse acesso não tem a condição de exigir HTTPS para as operações.

Em outras palavras, qualquer conta que tenha acesso entre contas pode acessar `DeleteObject`, `GetObject` e assim por diante, mas não `PutObject`, e pode fazer isso por HTTP ou HTTPS.

Se você não excluir a operação PutObject da segunda instrução, a instrução não será válida, porque se você incluir PutObject, é necessário definir explicitamente o HTTPS como uma condição.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CrossAccountRolePostOverHttps",
      "Effect": "Allow",
      "Action": "mediastore:PutObject",
      "Principal": {
        "AWS": "arn:aws:iam::<other acct number>:role/<role name>"
      },
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container name>/*",
      "Condition": {
        "Bool": {
          "aws:SecureTransport": "true"
        }
      }
    },
    {
      "Sid": "CrossAccountFullAccessExceptPost",
      "Effect": "Allow",
      "NotAction": "mediastore:PutObject",
      "Principal": {
        "AWS": "arn:aws:iam::<other acct number>:role/<role name>"
      },
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/<container name>/*"
    }
  ]
}
```

Exemplo de política de contêiner: acesso restrito a endereços IP específicos

Este exemplo de política permite o acesso a todas as operações do AWS Elemental MediaStore em objetos no contêiner especificado. No entanto, a solicitação deve se originar no intervalo de endereços IP especificados na condição.

A condição nesta instrução identifica o intervalo 198.51.100.* de endereços IP do protocolo de internet versão 4 (IPv4), com uma exceção: 198.51.100.188.

O bloco da Condition usa as condições IpAddress e NotIpAddress e a chave de condição aws:SourceIp, que é uma chave de condição que abrange toda a AWS. Os valores IPv4 aws:sourceIp usam a notação CIDR padrão. Para obter mais informações, consulte [Operadores de condição de endereço IP](#) no Guia do usuário do IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AccessBySpecificIPAddress",
      "Effect": "Allow",
      "Action": [
        "mediastore:GetObject",
        "mediastore:DescribeObject"
      ],
      "Principal": "*",
      "Resource": "arn:aws:mediastore:<region>:<owner acct number>:container/
<container name>/*",
      "Condition": {
        "IpAddress": {
          "aws:SourceIp": [
            "198.51.100.0/24"
          ]
        },
        "NotIpAddress": {
          "aws:SourceIp": "198.51.100.188/32"
        }
      }
    }
  ]
}
```

Políticas de compartilhamento de recursos de origem cruzada (CORS) no AWS Elemental MediaStore

O compartilhamento de recursos de origem cruzada (CORS) define uma maneira de os aplicativos web clientes carregados em um domínio interagirem com recursos em outro domínio. Com suporte

ao CORS no AWS Elemental MediaStore, você pode criar aplicativos web avançados do lado do cliente com o MediaStore e permitir seletivamente o acesso de origem cruzada aos recursos do MediaStore.

Note

Se você estiver usando o Amazon CloudFront para distribuir o conteúdo de um contêiner que tem uma política de CORS, [configure a distribuição para o AWS Elemental MediaStore](#) (incluindo a etapa para editar o comportamento de cache para configurar o CORS).

Esta seção fornece uma visão geral do CORS. Os subtópicos descrevem como você pode habilitar o CORS usando o console do AWS Elemental MediaStore ou, programaticamente, usando a API REST do MediaStore e os SDKs da AWS.

Tópicos

- [Cenários de casos de uso do CORS](#)
- [Adicionar uma política de CORS a um contêiner](#)
- [Visualizar uma política de CORS](#)
- [Editar uma política de CORS](#)
- [Excluir uma política de CORS](#)
- [Solução de problemas de CORS](#)
- [Exemplos de políticas de CORS](#)

Cenários de casos de uso do CORS

Veja a seguir exemplos de cenário de uso do CORS:

- **Cenário 1:** Vamos supor que você esteja distribuindo streaming de vídeo ao vivo em um contêiner do AWS Elemental MediaStore chamado LiveVideo. Seus usuários carregam o endpoint de manifesto do vídeo `http://livevideo.mediastore.ap-southeast-2.amazonaws.com` de uma origem específica, como `www.example.com`. Você deseja usar um reprodutor de vídeo JavaScript para acessar vídeos que são criados a partir desse contêiner por meio de solicitações GET e PUT não autenticadas. Normalmente, um navegador impediria que o JavaScript permitisse essas solicitações, mas você pode definir uma política de CORS em seu contêiner para permitir explicitamente essas solicitações em `www.example.com`.

- Cenário 2: suponha que você queira hospedar o mesmo streaming ao vivo, como no Cenário 1, do contêiner do MediaStore, mas deseja permitir solicitações de qualquer origem. Você pode configurar uma política de CORS para permitir origens curinga (*), para que solicitações de qualquer origem possam acessar o vídeo.

Adicionar uma política de CORS a um contêiner

Esta seção explica como adicionar uma configuração de compartilhamento de recursos de origem cruzada (CORS) a um contêiner do AWS Elemental MediaStore. O CORS permite que aplicativos web do cliente que sejam carregados em um domínio interajam em outro domínio.

Para configurar seu contêiner para permitir solicitações de origem cruzada, adicione uma política de CORS ao contêiner. Uma política de CORS define regras que identificam as origens que você permite que acessem seu contêiner, as operações (métodos HTTP) compatíveis para cada origem e outras informações específicas da operação.

Quando você adiciona uma política de CORS ao contêiner, as [políticas de contêiner](#) (que controlam os direitos de acesso ao contêiner), continuam a ser aplicadas.

Para adicionar uma política de CORS (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner para o qual você deseja criar uma política de CORS.

A página de detalhes do contêiner é exibida.

3. Na seção Container CORS policy (Política de CORS de contêiner), selecione Create CORS policy (Criar política de CORS).
4. Insira a política no formato JSON e, em seguida, selecione Save (Salvar).

Para adicionar uma política de CORS (AWS CLI)

1. Crie um arquivo que defina a política de CORS:

```
[
  {
    "AllowedHeaders": [
      "*"
    ]
  }
]
```



```
    ],
    "AllowedMethods": [
      "GET",
      "HEAD"
    ],
    "AllowedOrigins": [
      "*"
    ],
    "MaxAgeSeconds": 3000
  }
]
```

2. Na AWS CLI, use o comando `put-cors-policy`.

```
aws mediastore put-cors-policy --container-name ExampleContainer --cors-policy
file://corsPolicy.json --region us-west-2
```

Este comando não possui valor de retorno.

Visualizar uma política de CORS

O compartilhamento de recursos de origem cruzada (CORS) define uma maneira de os aplicativos web clientes carregados em um domínio interagirem com recursos em outro domínio.

Para visualizar uma política de CORS (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner para o qual você deseja visualizar a política de CORS.

A página de detalhes do contêiner será exibida, com a política de CORS na seção Container CORS policy (Política de CORS de contêiner).

Para visualizar uma política de CORS (AWS CLI)

- Na AWS CLI, use o comando `get-cors-policy`:

```
aws mediastore get-cors-policy --container-name ExampleContainer --region us-west-2
```

O exemplo a seguir mostra o valor de retorno:

```
{
  "CorsPolicy": [
    {
      "AllowedMethods": [
        "GET",
        "HEAD"
      ],
      "MaxAgeSeconds": 3000,
      "AllowedOrigins": [
        "*"
      ],
      "AllowedHeaders": [
        "*"
      ]
    }
  ]
}
```

Editar uma política de CORS

O compartilhamento de recursos de origem cruzada (CORS) define uma maneira de os aplicativos web clientes carregados em um domínio interagirem com recursos em outro domínio.

Para editar uma política de CORS (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner para o qual você deseja editar a política de CORS.

A página de detalhes do contêiner é exibida.

3. Na seção Container CORS policy (Política de CORS de contêiner), selecione Edit CORS policy (Editar política de CORS).
4. Faça as alterações na política e selecione Save (Salvar).

Para editar uma política de CORS (AWS CLI)

1. Crie um arquivo que defina a política CORS atualizada:

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "HEAD"
    ],
    "AllowedOrigins": [
      "https://www.example.com"
    ],
    "MaxAgeSeconds": 3000
  }
]
```

2. Na AWS CLI, use o comando `put-cors-policy`.

```
aws mediastore put-cors-policy --container-name ExampleContainer --cors-policy
file://corsPolicy2.json --region us-west-2
```

Este comando não possui valor de retorno.

Excluir uma política de CORS

O compartilhamento de recursos de origem cruzada (CORS) define uma maneira de os aplicativos web clientes carregados em um domínio interagirem com recursos em outro domínio. Excluir a política de CORS de um contêiner remove permissões para solicitações de origem cruzada.

Para excluir uma política de CORS (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner para o qual você deseja excluir a política de CORS.

A página de detalhes do contêiner é exibida.

3. Na seção Container CORS policy (Política de CORS de contêiner), selecione Delete CORS policy (Excluir política de CORS).
4. Escolha Continue (Continuar) para confirmar e escolha Save (Salvar).

Para excluir uma política de CORS (AWS CLI)

- Na AWS CLI, use o comando `delete-cors-policy`:

```
aws mediastore delete-cors-policy --container-name ExampleContainer --region us-west-2
```

Este comando não possui valor de retorno.

Solução de problemas de CORS

Se você notar um comportamento inesperado ao acessar um contêiner que tem uma política de CORS, siga estas etapas para solucionar o problema.

1. Verifique se a política de CORS está anexada ao contêiner.

Para obter instruções, consulte [the section called “Visualizar uma política de CORS”](#).

2. Capture a solicitação e a resposta completas usando uma ferramenta de sua escolha, como o console do desenvolvedor do navegador. Verifique se a política de CORS anexada ao contêiner inclui pelo menos uma regra de CORS que corresponda aos dados na solicitação, da seguinte maneira:
 - a. Verifique se a solicitação tem um cabeçalho `Origin`.

Se o cabeçalho estiver ausente, o AWS Elemental MediaStore não tratará a solicitação como uma solicitação de origem cruzada e não enviará cabeçalhos de resposta de CORS de volta na resposta.

- b. Verifique se o cabeçalho `Origin` na solicitação corresponde a pelo menos um dos elementos `AllowedOrigins` na `CORSRule` específica.

O esquema, o host e os valores de porta no cabeçalho da solicitação `Origin` devem corresponder a `AllowedOrigins` na `CORSRule`. Por exemplo, se você definir a `CORSRule` para permitir a origem `http://www.example.com`, as origens `https://`

`www.example.com` e `http://www.example.com:80` da solicitação não corresponderão à origem permitida na configuração.

- c. Verifique se o método na solicitação (ou o método especificado em `Access-Control-Request-Method` no caso de uma solicitação de simulação) é um dos elementos `AllowedMethods` na mesma `CORSRule`.
- d. Para uma solicitação de simulação, se a solicitação incluir um cabeçalho `Access-Control-Request-Headers`, verifique se `CORSRule` inclui as entradas `AllowedHeaders` para cada valor no cabeçalho `Access-Control-Request-Headers` header.

Exemplos de políticas de CORS

Os exemplos a seguir mostram as políticas de compartilhamento de recursos de origem cruzada (CORS).

Tópicos

- [Exemplo de política de CORS: acesso de leitura para qualquer domínio](#)
- [Exemplo de política de CORS: acesso de leitura para um domínio específico](#)

Exemplo de política de CORS: acesso de leitura para qualquer domínio

A política a seguir permite que uma página da web de qualquer domínio recupere o conteúdo do contêiner do AWS Elemental MediaStore. A solicitação inclui todos os cabeçalhos HTTP do domínio de origem, e o serviço responde apenas às solicitações HTTP GET e HTTP HEAD do domínio de origem. Os resultados são armazenados em cache por 3.000 segundos antes de um novo conjunto de resultados ser fornecido.

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "HEAD"
    ],
    "AllowedOrigins": [
```

```
    "*"
  ],
  "MaxAgeSeconds": 3000
}
]
```

Exemplo de política de CORS: acesso de leitura para um domínio específico

A política a seguir permite que uma página da web de `https://www.example.com` recupere o conteúdo do contêiner do AWS Elemental MediaStore. A solicitação inclui todos os cabeçalhos HTTP do `https://www.example.com`, e o serviço responde apenas às solicitações HTTP GET e HTTP HEAD do `https://www.example.com`. Os resultados são armazenados em cache por 3.000 segundos antes de um novo conjunto de resultados ser fornecido.

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "HEAD"
    ],
    "AllowedOrigins": [
      "https://www.example.com"
    ],
    "MaxAgeSeconds": 3000
  }
]
```

Políticas de ciclo de vida de objetos no AWS Elemental MediaStore

Para cada contêiner, você pode criar uma política de ciclo de vida de objetos que controla por quanto tempo os objetos devem ser armazenados no contêiner. Quando os objetos chegarem à idade máxima especificada, o AWS Elemental MediaStore excluirá os objetos. Você pode excluir objetos quando eles não forem mais necessários para economizar custos de armazenamento.

Também é possível especificar que o MediaStore deve mover objetos para a classe de armazenamento de acesso infrequente (IA) depois que eles atingem uma determinada idade. Os objetos armazenados na classe de armazenamento IA têm taxas de armazenamento e recuperação

diferentes dos objetos armazenados na classe de armazenamento padrão. Para obter mais informações, consulte [Preços do MediaStore](#).

Uma política de ciclo de vida de objetos contém regras, que determinam o ciclo de vida de objetos por subpasta. (Você não pode atribuir uma política de ciclo de vida de objetos a objetos individuais). Você pode anexar apenas uma política de ciclo de vida de objetos a um contêiner, mas você pode adicionar até 10 regras a cada política de ciclo de vida de objeto. Para obter mais informações, consulte [Componentes de uma política de ciclo de vida de objetos](#).

Tópicos

- [Componentes de uma política de ciclo de vida de objetos](#)
- [Adicionar uma política de ciclo de vida de objetos a um contêiner](#)
- [Visualizar uma política de ciclo de vida de objetos](#)
- [Editar uma política de ciclo de vida de objetos](#)
- [Excluir uma política de ciclo de vida de objetos](#)
- [Exemplos de política de ciclo de vida de objetos](#)

Componentes de uma política de ciclo de vida de objetos

As políticas de ciclo de vida de objeto controlam quanto tempo os objetos permanecem em um contêiner do AWS Elemental MediaStore. Cada política de ciclo de vida de objetos consiste em uma ou mais regras, que determinam o ciclo de vida de objetos. Uma regra pode ser aplicada a uma pasta, várias pastas ou a todo o contêiner.

Você pode anexar uma política de ciclo de vida de objeto a um contêiner, e cada política de ciclo de vida de objetos pode conter até 10 regras. Você não pode atribuir uma política de ciclo de vida de objetos a um único objeto.

Regras em uma política de ciclo de vida de objetos

É possível criar três tipos de regras:

- [Dados temporários](#)
- [Exclusão de objeto](#)
- [Transição do ciclo de vida](#)

Dados temporários

Uma regra de dados temporários define objetos para expirarem em segundos. Esse tipo de regra se aplica somente a objetos adicionados ao contêiner após a diretiva entrar em vigor. Leva até 20 minutos para o MediaStore aplicar a nova política ao contêiner.

Um exemplo de uma regra para dados temporários é semelhante ao seguinte:

```
{
  "definition": {
    "path": [ {"wildcard": "Football/index*.m3u8"} ],
    "seconds_since_create": [
      {"numeric": [ ">", 120 ]}
    ]
  },
  "action": "EXPIRE"
},
```

As regras de dados temporários têm três partes:

- **path**: sempre definido como **wildcard**. Use essa parte para definir quais objetos deseja excluir. É possível usar um ou mais curingas, representados por um asterisco (*). Cada curinga representa qualquer combinação de zero ou mais caracteres. Por exemplo, "path": [{"wildcard": "Football/index*.m3u8"}], aplica-se a todos os arquivos na pasta Football que correspondem ao padrão de index*.m3u8 (como index.m3u8, index1.m3u8 e index123456.m3u8). É possível incluir até 10 caminhos em uma única regra.
- **seconds_since_create**: sempre definido como **numeric**. É possível especificar um valor de 1 a 300 segundos. Também é possível definir o operador como "maior que" (>) ou "maior que ou igual a" (>=).
- **action**: sempre definido como EXPIRE.

Para regras de dados temporários (objetos expiram em segundos), não há atraso entre a expiração de um objeto e a exclusão do objeto.

Note

Os objetos que estão sujeitos a uma regra de dados temporários não são incluídos em uma resposta `list-items`. Além disso, objetos que expiram devido a uma regra de dados transitória não emitem um evento do CloudWatch quando expiram.

Exclusão de objeto

Uma regra de exclusão de objeto define objetos para expirar em dias. Esse tipo de regra se aplica a todos os objetos do contêiner, mesmo que tenham sido adicionados a ele antes da criação da política. Leva até 20 minutos para que o MediaStore aplique a nova política, mas pode levar até 24 horas para que os objetos sejam limpos do contêiner.

Um exemplo de duas regras para excluir objetos é semelhante ao seguinte:

```
{
  "definition": {
    "path": [ { "prefix": "FolderName/" } ],
    "days_since_create": [
      {"numeric": [ ">" , 5]}
    ]
  },
  "action": "EXPIRE"
},
{
  "definition": {
    "path": [ { "wildcard": "Football/*.ts" } ],
    "days_since_create": [
      {"numeric": [ ">" , 5]}
    ]
  },
  "action": "EXPIRE"
}
```

As regras de exclusão de objetos têm três partes:

- `path`: defina como `prefix` ou `wildcard`. Você não pode misturar `prefix` e `wildcard` na mesma regra. Se quiser usar ambos, será necessário criar uma regra para `prefix` e uma regra separada para `wildcard`, como mostrado no exemplo acima.

- `prefix` – defina o caminho para `prefix` se quiser excluir todos os objetos de uma pasta específica. Se o parâmetro estiver vazio (`"path": [{ "prefix": "" }],`), o destino será todos os objetos armazenados em qualquer lugar no contêiner atual. É possível incluir até 10 caminhos `prefix` em uma única regra.
- `wildcard` – defina o caminho para `wildcard` se quiser excluir objetos específicos com base no nome de arquivo e/ou tipo de arquivo. É possível usar um ou mais curingas, representados por um asterisco (*). Cada curinga representa qualquer combinação de zero ou mais caracteres. Por exemplo, `"path": [{"wildcard": "Football/*.ts"}]`, aplica-se a todos os arquivos na pasta `Football` que correspondem ao padrão de `*.ts` (como `filename.ts`, `filename1.ts` e `filename123456.ts`). É possível incluir até 10 caminhos `wildcard` em uma única regra.
- `days_since_create`: sempre definido como `numeric`. É possível especificar um valor de 1 a 36.500 dias. Também é possível definir o operador como “maior que” (`>`) ou “maior que ou igual a” (`>=`).
- `action`: sempre definido como `EXPIRE`.

Para excluir regras de objetos (objetos expiram em dias), pode haver um pequeno atraso entre a expiração de um objeto e a exclusão do objeto. No entanto, as alterações no faturamento ocorrem assim que o objeto expira. Por exemplo, se uma regra de ciclo de vida especificar `10 days_since_create`, a conta não será cobrada pelo objeto depois que o objeto tiver 10 dias, mesmo que ele ainda não tenha sido excluído.

Transição do ciclo de vida

Uma regra de transição do ciclo de vida define os objetos a serem movidos para a classe de armazenamento de acesso infrequente (IA) depois que eles atingem uma determinada idade, medida em dias. Os objetos armazenados na classe de armazenamento IA têm taxas de armazenamento e recuperação diferentes dos objetos armazenados na classe de armazenamento padrão. Para obter mais informações, consulte [Preços do MediaStore](#).

Depois que um objeto for movido para a classe de armazenamento IA, não será possível movê-lo de volta para a classe de armazenamento padrão.

A regra de transição do ciclo de vida se aplica a todos os objetos do contêiner, mesmo que tenham sido adicionados a ele antes da criação da política. Leva até 20 minutos para que o MediaStore aplique a nova política, mas pode levar até 24 horas para que os objetos sejam limpos do contêiner.

Veja a seguir um exemplo de regra de transição do ciclo de vida:

```
{
  "definition": {
    "path": [
      {"prefix": "AwardsShow/"}
    ],
    "days_since_create": [
      {"numeric": [">=" , 30]}
    ]
  },
  "action": "ARCHIVE"
}
```

As regras de transição do ciclo de vida têm três partes:

- **path**: defina como **prefix** ou **wildcard**. Você não pode misturar **prefix** e **wildcard** na mesma regra. Se você quiser usar ambos, deverá criar uma regra para **prefix** e uma regra separada para **wildcard**.
- **prefix** – defina o caminho como **prefix** se desejar fazer a transição de todos os objetos em uma pasta específica para a classe de armazenamento IA. Se o parâmetro estiver vazio ("path": [{ "prefix": "" }],), o destino será todos os objetos armazenados em qualquer lugar no contêiner atual. É possível incluir até 10 caminhos **prefix** em uma única regra.
- **wildcard** – defina o caminho como **wildcard** se desejar fazer a transição de objetos específicos para a classe de armazenamento IA com base no nome do arquivo e/ou no tipo de arquivo. É possível usar um ou mais curingas, representados por um asterisco (*). Cada curinga representa qualquer combinação de zero ou mais caracteres. Por exemplo, "path": [{"wildcard": "Football/*.ts"}], aplica-se a todos os arquivos na pasta `Football` que correspondem ao padrão de `*.ts` (como `filename.ts`, `filename1.ts` e `filename123456.ts`). É possível incluir até 10 caminhos **wildcard** em uma única regra.
- **days_since_create**: sempre definido como "numeric": [">=" , 30].
- **action**: sempre definido como `ARCHIVE`.

Exemplo

Vamos supor que um contêiner chamado `LiveEvents` tenha quatro subpastas: `Football`, `Baseball`, `Basketball` e `AwardsShow`. A política de ciclo de vida de objetos atribuída à pasta `LiveEvents` pode ter a seguinte aparência:

```

{
  "rules": [
    {
      "definition": {
        "path": [
          {"prefix": "Football/"},
          {"prefix": "Baseball/"}
        ],
        "days_since_create": [
          {"numeric": [ ">" , 28]}
        ]
      },
      "action": "EXPIRE"
    },
    {
      "definition": {
        "path": [ { "prefix": "AwardsShow/" } ],
        "days_since_create": [
          {"numeric": [ ">=" , 15]}
        ]
      },
      "action": "EXPIRE"
    },
    {
      "definition": {
        "path": [ { "prefix": "" } ],
        "days_since_create": [
          {"numeric": [ ">" , 40]}
        ]
      },
      "action": "EXPIRE"
    },
    {
      "definition": {
        "path": [ { "wildcard": "Football/*.ts" } ],
        "days_since_create": [
          {"numeric": [ ">" , 20]}
        ]
      },
      "action": "EXPIRE"
    },
    {
      "definition": {

```

```

        "path": [
            {"wildcard": "Football/index*.m3u8"}
        ],
        "seconds_since_create": [
            {"numeric": [">" , 15]}
        ]
    },
    "action": "EXPIRE"
},
{
    "definition": {
        "path": [
            {"prefix": "Program/" }
        ],
        "days_since_create": [
            {"numeric": [">=" , 30]}
        ]
    },
    "action": "ARCHIVE"
}
]
}

```

A política anterior especifica o seguinte:

- A primeira regra instrui o AWS Elemental MediaStore a excluir objetos armazenados na pasta LiveEvents/Football e na pasta LiveEvents/Baseball depois de completarem 28 dias.
- A segunda regra instrui o serviço a excluir objetos armazenados na pasta LiveEvents/AwardsShow ao completarem 15 dias ou mais.
- A terceira regra instrui o serviço a excluir objetos armazenados em qualquer lugar no contêiner LiveEvents depois de completarem 40 dias. Essa regra se aplica a objetos armazenados diretamente no contêiner LiveEvents, bem como objetos armazenados em qualquer uma das quatro subpastas do contêiner.
- A quarta regra instrui o serviço a excluir objetos na pasta Football que correspondem ao padrão *.ts depois que tiverem mais de 20 dias.
- A quinta regra instrui o serviço a excluir objetos na pasta Football que correspondem ao padrão index*.m3u8 depois que tiverem mais de 15 segundos. O MediaStore exclui esses arquivos 16 segundos depois de eles serem colocados no contêiner.

- A sexta regra instrui o serviço a mover objetos na pasta Program para a classe de armazenamento IA após 30 dias de idade.

Para obter mais exemplos de políticas de ciclo de vida de objetos, consulte [Exemplos de política de ciclo de vida de objetos](#).

Adicionar uma política de ciclo de vida de objetos a um contêiner

Uma política de ciclo de vida de objetos permite especificar por quanto tempo armazenar seus objetos em um contêiner. Você define uma data de validade e, depois dessa data, o AWS Elemental MediaStore exclui os objetos. Leva até 20 minutos para o serviço aplicar a nova política ao contêiner.

Para obter informações sobre como criar uma política de ciclo de vida, consulte [Componentes de uma política de ciclo de vida de objetos](#).

Note

Para excluir regras de objetos (objetos expiram em dias), pode haver um pequeno atraso entre a expiração de um objeto e a exclusão do objeto. No entanto, as alterações no faturamento ocorrem assim que o objeto expira. Por exemplo, se uma regra de ciclo de vida especificar `10 days_since_create`, a conta não será cobrada pelo objeto depois que o objeto tiver 10 dias, mesmo que ele ainda não tenha sido excluído.

Como adicionar uma política de ciclo de vida de objetos (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner para o qual você deseja criar uma política de ciclo de vida de objetos.

A página de detalhes do contêiner é exibida.

3. Na seção Object lifecycle policy (Política de ciclo de vida de objetos) escolha Create object lifecycle policy (Criar política de ciclo de vida de objetos).
4. Insira a política no formato JSON e, em seguida, selecione Save (Salvar).

Como adicionar uma política de ciclo de vida de objetos (AWS CLI)

1. Crie um arquivo que define a política de ciclo de vida de objetos:

```
{
  "rules": [
    {
      "definition": {
        "path": [
          {"prefix": "Football/"},
          {"prefix": "Baseball/"}
        ],
        "days_since_create": [
          {"numeric": [ ">" , 28]}
        ]
      },
      "action": "EXPIRE"
    },
    {
      "definition": {
        "path": [
          {"wildcard": "AwardsShow/index*.m3u8"}
        ],
        "seconds_since_create": [
          {"numeric": [ ">" , 8]}
        ]
      },
      "action": "EXPIRE"
    }
  ]
}
```

2. Na AWS CLI, use o comando `put-lifecycle-policy`:

```
aws mediastore put-lifecycle-policy --container-name LiveEvents --lifecycle-policy file://LiveEventsLifecyclePolicy.json --region us-west-2
```

Este comando não possui valor de retorno. O serviço anexa a política especificada ao contêiner.

Visualizar uma política de ciclo de vida de objetos

Uma política de ciclo de vida de objetos especifica por quanto tempo os objetos devem ser mantidos em um contêiner.

Como visualizar uma política de ciclo de vida de objetos (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner para o qual você deseja visualizar a política de ciclo de vida de objetos.

A página de detalhes do contêiner é exibida, com a política de ciclo de vida de objetos na seção Object lifecycle policy (Política de ciclo de vida de objetos).

Para visualizar uma política de ciclo de vida de objetos (AWS CLI)

- Na AWS CLI, use o comando `get-lifecycle-policy`:

```
aws mediastore get-lifecycle-policy --container-name LiveEvents --region us-west-2
```

O exemplo a seguir mostra o valor de retorno:

```
{
  "LifecyclePolicy": "{
    "rules": [
      {
        "definition": {
          "path": [
            {"prefix": "Football/"},
            {"prefix": "Baseball/"}
          ],
          "days_since_create": [
            {"numeric": [">" , 28]}
          ]
        },
        "action": "EXPIRE"
      }
    ]
  }"
```


Editar uma política de ciclo de vida de objetos

Você não pode editar uma política de ciclo de vida de objetos existente. No entanto, você pode alterar uma política existente fazendo upload de uma política de substituição. Leva até 20 minutos para o serviço aplicar a política atualizada ao contêiner.

Como editar uma política de ciclo de vida de objetos (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner para o qual você deseja editar a política de ciclo de vida de objetos.

A página de detalhes do contêiner é exibida.

3. Na seção Object lifecycle policy (Política de ciclo de vida de objetos), escolha Edit object lifecycle policy (Editar política de ciclo de vida de objetos).
4. Faça as alterações na política e selecione Save (Salvar).

Como editar uma política de ciclo de vida de objetos (AWS CLI)

1. Crie um arquivo que define a política de ciclo de vida do objeto atualizada:

```
{
  "rules": [
    {
      "definition": {
        "path": [
          {"prefix": "Football/"},
          {"prefix": "Baseball/"},
          {"prefix": "Basketball/"},
        ],
        "days_since_create": [
          {"numeric": [">" , 28]}
        ]
      },
      "action": "EXPIRE"
    }
  ]
}
```

2. Na AWS CLI, use o comando `put-lifecycle-policy`:

```
aws mediastore put-lifecycle-policy --container-name LiveEvents --lifecycle-policy file://LiveEvents2LifecyclePolicy --region us-west-2
```

Este comando não possui valor de retorno. O serviço anexa a política especificada ao contêiner, substituindo a política anterior.

Excluir uma política de ciclo de vida de objetos

Quando você exclui uma política de ciclo de vida de objeto, leva até 20 minutos para o serviço aplicar a alteração ao contêiner.

Como excluir uma política de ciclo de vida de objetos (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner para o qual você deseja excluir a política de ciclo de vida de objetos.

A página de detalhes do contêiner é exibida.

3. Na seção Object lifecycle policy (Política de ciclo de vida de objetos), escolha Delete lifecycle policy (Excluir política de ciclo de vida de objetos).
4. Escolha Continue (Continuar) para confirmar e escolha Save (Salvar).

Para excluir uma política de ciclo de vida de objetos (AWS CLI)

- Na AWS CLI, use o comando `delete-lifecycle-policy`:

```
aws mediastore delete-lifecycle-policy --container-name LiveEvents --region us-west-2
```

Este comando não possui valor de retorno.

Exemplos de política de ciclo de vida de objetos

Os exemplos a seguir mostram políticas de ciclo de vida de objetos.

Tópicos

- [Exemplo de política de ciclo de vida de objetos: Expirar em segundos](#)
- [Exemplo de política de ciclo de vida de objetos: Expirar em dias](#)
- [Exemplo de política de ciclo de vida de objetos: transição para classe de armazenamento de acesso infrequente](#)
- [Exemplo de política de ciclo de vida de objetos: Várias regras](#)
- [Exemplo de política de ciclo de vida de objetos: Esvaziar contêiner](#)

Exemplo de política de ciclo de vida de objetos: Expirar em segundos

A política a seguir especifica que o MediaStore exclua objetos que correspondam a todos os seguintes critérios:

- O objeto é adicionado ao contêiner após a política entrar em vigor.
- O objeto é armazenado na pasta Football.
- O objeto tem uma extensão de arquivo de m3u8.
- O objeto está no contêiner por mais de 20 segundos.

```
{
  "rules": [
    {
      "definition": {
        "path": [
          {"wildcard": "Football/*.m3u8"}
        ],
        "seconds_since_create": [
          {"numeric": [ ">", 20 ]}
        ]
      },
      "action": "EXPIRE"
    }
  ]
}
```

Exemplo de política de ciclo de vida de objetos: Expirar em dias

A política a seguir especifica que o MediaStore exclua objetos que correspondam a todos os seguintes critérios:

- O objeto é armazenado na pasta Program
- O objeto tem uma extensão de arquivo de ts
- O objeto está no contêiner há mais de 5 dias

```
{
  "rules": [
    {
      "definition": {
        "path": [
          {"wildcard": "Program/*.ts"}
        ],
        "days_since_create": [
          {"numeric": [ ">", 5 ]}
        ]
      },
      "action": "EXPIRE"
    }
  ]
}
```

Exemplo de política de ciclo de vida de objetos: transição para classe de armazenamento de acesso infrequente

A política a seguir especifica que o MediaStore mova objetos para a classe de armazenamento de acesso infrequente (IA) quando eles tiverem 30 dias de idade. Os objetos armazenados na classe de armazenamento IA têm taxas de armazenamento e recuperação diferentes dos objetos armazenados na classe de armazenamento padrão.

O campo `days_since_create` deve ser definido como `"numeric": [">=" , 30]`.

```
{
  "rules": [
    {
      "definition": {
        "path": [
          {"prefix": "Football/"},
          {"prefix": "Baseball/"}
        ],
        "days_since_create": [
          {"numeric": [ ">=" , 30 ]}
        ]
      }
    }
  ]
}
```

```

    ]
  },
  "action": "ARCHIVE"
}
]
}

```

Exemplo de política de ciclo de vida de objetos: Várias regras

A política a seguir especifica que o MediaStore faça o seguinte:

- Mova objetos armazenados na pasta AwardsShow para a classe de armazenamento de acesso infrequente (IA) após 30 dias
- Exclua objetos com a extensão de arquivo m3u8 e que estejam armazenados na pasta Football após 20 segundos
- Exclua objetos armazenados na pasta April após 10 dias
- Exclua objetos com a extensão de arquivo ts e que estejam armazenados na pasta Program após 5 dias

```

{
  "rules": [
    {
      "definition": {
        "path": [
          {"prefix": "AwardsShow/"}
        ],
        "days_since_create": [
          {"numeric": [ ">=" , 30 ]}
        ]
      },
      "action": "ARCHIVE"
    },
    {
      "definition": {
        "path": [
          {"wildcard": "Football/*.m3u8"}
        ],
        "seconds_since_create": [
          {"numeric": [ ">", 20 ]}
        ]
      }
    }
  ]
}

```

```

    },
    "action": "EXPIRE"
  },
  {
    "definition": {
      "path": [
        {"prefix": "April"}
      ],
      "days_since_create": [
        {"numeric": [ ">", 10 ]}
      ]
    },
    "action": "EXPIRE"
  },
  {
    "definition": {
      "path": [
        {"wildcard": "Program/*.ts"}
      ],
      "days_since_create": [
        {"numeric": [ ">", 5 ]}
      ]
    },
    "action": "EXPIRE"
  }
]
}

```

Exemplo de política de ciclo de vida de objetos: Esvaziar contêiner

A política de ciclo de vida de objetos a seguir especifica que o MediaStore exclua todos os objetos no contêiner, inclusive pastas e subpastas, 1 dia após serem adicionados ao contêiner. Se o contêiner reter quaisquer objetos antes que essa política seja aplicada, o MediaStore excluirá os objetos 1 dia após a política entrar em vigor. Leva até 20 minutos para o serviço aplicar a nova política ao contêiner.

```

{
  "rules": [
    {
      "definition": {
        "path": [
          {"wildcard": "*"}
        ],

```

```
        "days_since_create": [
            {"numeric": [ ">=", 1 ]}
        ],
        "action": "EXPIRE"
    }
]
```

Políticas de métrica no AWS Elemental MediaStore

Para cada contêiner, você pode adicionar uma política de métrica para permitir que o AWS Elemental MediaStore envie métricas para o Amazon CloudWatch. Pode demorar até 20 minutos para que a nova política tenha efeito. Para obter uma descrição de cada métrica do MediaStore, consulte [Métricas do MediaStore](#).

Uma política de métrica contém o seguinte:

- Uma configuração para habilitar ou desabilitar métricas no nível do contêiner.
- De zero a cinco regras que habilitam métricas no nível do objeto. Se a política contiver regras, cada regra deverá incluir ambas as seguintes opções:
 - Um grupo de objetos que define quais objetos devem ser incluídos no grupo. A definição pode ser um caminho ou um nome de arquivo, mas não pode ter mais de 900 caracteres. Os caracteres válidos são: a–z, A–Z, 0–9, _ (sublinhado), = (igual), : (dois-pontos), . (ponto), - (hífen), ~ (til), / (barra) e * (asterisco). Curingas (*) são aceitáveis.
 - Um nome de grupo de objetos que permita fazer referência ao grupo de objetos. O nome não pode ter mais de 30 caracteres. Os caracteres válidos são a–z, A–Z, 0–9 e _ (sublinhado).

Se um objeto corresponder a várias regras, o CloudWatch exibirá um ponto de dados para cada regra correspondente. Por exemplo, se um objeto corresponder a duas regras chamadas `rule1` e `rule2`, o CloudWatch exibirá dois pontos de dados para essas regras. O primeiro tem uma dimensão de `ObjectGroupName=rule1` e o segundo tem uma dimensão de `ObjectGroupName=rule2`.

Tópicos

- [Adicionar uma política de métrica](#)
- [Visualizar uma política de métrica](#)

- [Editar uma política de métrica](#)
- [Exemplos de políticas de métrica](#)

Adicionar uma política de métrica

Uma política de métrica contém regras que ditam quais métricas o AWS Elemental MediaStore envia para o Amazon CloudWatch. Para obter exemplos de políticas de métrica, consulte [Exemplos de políticas de métrica](#).

Como adicionar uma política de métrica (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), escolha o nome do contêiner ao qual você deseja adicionar uma política de métrica.

A página de detalhes do contêiner é exibida.

3. Na seção Metric policy (Política de métrica), escolha Create metric policy (Criar política de métrica).
4. Insira a política no formato JSON e, em seguida, selecione Save (Salvar).

Visualizar uma política de métrica

É possível usar o console ou a AWS CLI para visualizar a política de métrica de um contêiner.

Como visualizar uma política de métrica (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), escolha o nome do contêiner.

A página de detalhes do contêiner é exibida. A política é exibida na seção Metric policy (Política de métrica).

Editar uma política de métrica

Uma política de métrica contém regras que ditam quais métricas o AWS Elemental MediaStore envia ao Amazon CloudWatch. Quando você edita uma política de métrica existente, leva até 20 minutos

para que a nova política entre em vigor. Para obter exemplos de políticas de métrica, consulte [Exemplos de políticas de métrica](#).

Como editar uma política de métrica (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), escolha o nome do contêiner.
3. Na seção Metric policy (Política de métrica), escolha Edit metric policy (Editar política de métrica).
4. Faça as alterações apropriadas e, em seguida, selecione Save (Salvar).

Exemplos de políticas de métrica

Os exemplos a seguir mostram políticas de métrica criadas para diferentes casos de uso.

Tópicos

- [Exemplo de política de métrica: métricas em nível de contêiner](#)
- [Exemplo de política métrica: métricas em nível de caminho](#)
- [Exemplo de política de métrica: métricas em nível de contêiner e de caminho](#)
- [Exemplo de política métrica: métricas em nível de caminho usando curingas](#)
- [Exemplo de política métrica: métricas em nível de caminho com regras sobrepostas](#)

Exemplo de política de métrica: métricas em nível de contêiner

Este exemplo de política indica que o AWS Elemental MediaStore deve enviar métricas para o Amazon CloudWatch no nível de contêiner. Por exemplo, isso inclui a métrica RequestCount que conta o número de solicitações Put feitas ao contêiner. Como alternativa, é possível definir isso como DISABLED.

Como não há regras nesta política, o MediaStore não envia métricas no nível de caminho. Por exemplo, não é possível ver quantas solicitações Put foram feitas para uma pasta específica dentro desse contêiner.

```
{  
  "ContainerLevelMetrics": "ENABLED"
```

```
}
```

Exemplo de política métrica: métricas em nível de caminho

Este exemplo de política indica que o AWS Elemental MediaStore não deve enviar métricas para o Amazon CloudWatch no nível do contêiner. Além disso, o MediaStore deve enviar métricas para objetos em duas pastas específicas: `baseball/saturday` e `football/saturday`. As métricas para solicitações do MediaStore são as seguintes:

- As solicitações para a pasta `baseball/saturday` têm uma dimensão do CloudWatch de `ObjectGroupName=baseballGroup`.
- As solicitações para a pasta `football/saturday` têm uma dimensão `ObjectGroupName=footballGroup`.

```
{
  "ContainerLevelMetrics": "DISABLED",
  "MetricPolicyRules": [
    {
      "ObjectGroup": "baseball/saturday",
      "ObjectGroupName": "baseballGroup"
    },
    {
      "ObjectGroup": "football/saturday",
      "ObjectGroupName": "footballGroup"
    }
  ]
}
```

Exemplo de política de métrica: métricas em nível de contêiner e de caminho

Este exemplo de política indica que o AWS Elemental MediaStore deve enviar métricas para o Amazon CloudWatch no nível de contêiner. Além disso, o MediaStore deve enviar métricas para objetos em duas pastas específicas: `baseball/saturday` e `football/saturday`. As métricas para solicitações do MediaStore são as seguintes:

- As solicitações para a pasta `baseball/saturday` têm uma dimensão do CloudWatch de `ObjectGroupName=baseballGroup`.
- As solicitações para a pasta `football/saturday` têm uma dimensão `ObjectGroupName=footballGroup` do CloudWatch.

```
{
  "ContainerLevelMetrics": "ENABLED",
  "MetricPolicyRules": [
    {
      "ObjectGroup": "baseball/saturday",
      "ObjectGroupName": "baseballGroup"
    },
    {
      "ObjectGroup": "football/saturday",
      "ObjectGroupName": "footballGroup"
    }
  ]
}
```

Exemplo de política métrica: métricas em nível de caminho usando curingas

Este exemplo de política indica que o AWS Elemental MediaStore deve enviar métricas para o Amazon CloudWatch no nível de contêiner. Além disso, o MediaStore também deve enviar métricas para objetos com base em seus nomes de arquivo. Um caractere curinga indica que os objetos podem ser armazenados em qualquer lugar do contêiner e podem ter qualquer nome de arquivo, contanto que ele termine com uma extensão `.m3u8`.

```
{
  "ContainerLevelMetrics": "ENABLED",
  "MetricPolicyRules": [
    {
      "ObjectGroup": "*.m3u8",
      "ObjectGroupName": "index"
    }
  ]
}
```

Exemplo de política métrica: métricas em nível de caminho com regras sobrepostas

Este exemplo de política indica que o AWS Elemental MediaStore deve enviar métricas para o Amazon CloudWatch no nível de contêiner. Além disso, o MediaStore deve enviar métricas para duas pastas: `sports/football/saturday` e `sports/football`.

As métricas para solicitações do MediaStore à pasta `sports/football/saturday` têm uma dimensão do CloudWatch de `ObjectGroupName=footballGroup1`. Como os objetos armazenados na pasta `sports/football` correspondem a ambas as

regras, o CloudWatch exibe dois pontos de dados para esses objetos: um com uma dimensão de `ObjectGroupName=footballGroup1` e o segundo com uma dimensão de `ObjectGroupName=footballGroup2`.

```
{
  "ContainerLevelMetrics": "ENABLED",
  "MetricPolicyRules": [
    {
      "ObjectGroup": "sports/football/saturday",
      "ObjectGroupName": "footballGroup1"
    },
    {
      "ObjectGroup": "sports/football",
      "ObjectGroupName": "footballGroup2"
    }
  ]
}
```

Pastas no AWS Elemental MediaStore

As pastas são divisões dentro de um contêiner. Você usa as pastas para subdividir o contêiner da mesma maneira que você cria subpastas para dividir uma pasta em um sistema de arquivos. Você pode criar até 10 níveis de pastas, não incluindo o próprio contêiner.

As pastas são opcionais. Você pode optar por fazer upload de seus objetos diretamente para um contêiner em vez de uma pasta. No entanto, as pastas são uma maneira fácil de organizar os objetos.

Para fazer upload de um objeto para uma pasta, você especifica o caminho para a pasta. Se a pasta já existir, o AWS Elemental MediaStore armazenará o objeto na pasta. Se a pasta não existir, o serviço a cria e, em seguida, armazena o objeto na pasta.

Por exemplo, suponha que você tenha um contêiner chamado `movies` e faça upload de um arquivo chamado `m1aw.ts` com o caminho `premium/canada`. O AWS Elemental MediaStore armazena o objeto na subpasta "canada" sob a pasta "premium". Se nenhuma das pastas existir, o serviço criará a pasta `premium` e a subpasta `canada` e, em seguida, armazenará seu objeto na subpasta `canada`. Se você especificar apenas o contêiner `movies` (sem caminho), o serviço armazenará o objeto diretamente no contêiner.

O AWS Elemental MediaStore exclui automaticamente uma pasta quando você exclui o último objeto nela. O serviço também exclui todas as pastas vazias acima dessa pasta. Por exemplo, suponha que você tenha uma pasta chamada `premium` que não contenha nenhum arquivo, mas contenha uma subpasta denominada `canada`. A subpasta `canada` contém um arquivo chamado `m1aw.ts`. Se você excluir o arquivo `m1aw.ts`, o serviço excluirá as pastas `premium` e `canada`. Essa exclusão automática se aplica apenas a pastas. O serviço não exclui contêineres vazios.

Tópicos

- [Regras para nomes de pastas](#)
- [Criar uma pasta](#)
- [Excluir uma pasta](#)

Regras para nomes de pastas

Ao escolher um nome para a pasta, lembre-se do seguinte:

- O nome pode conter apenas os seguintes caracteres: letras maiúsculas (A-Z), letras minúsculas (a-z), números (0-9), pontos finais (.), hífens (-), tildes (~), sublinhados (_), sinais de igualdade (=) e dois-pontos (:).
- O nome deve ter pelo menos um caractere. Nomes de pastas vazias (como `folder1//folder3/`) não são permitidos.
- Os nomes diferenciam letras maiúsculas de minúsculas. Por exemplo, você pode ter uma pasta chamada `myFolder` e outra chamada `myfolder` no mesmo contêiner ou pasta porque esses nomes são exclusivos.
- O nome deve ser exclusivo apenas dentro do contêiner ou da pasta pai. Por exemplo, você pode criar uma pasta chamada `myfolder` em dois contêineres diferentes: `movies/myfolder` e `sports/myfolder`.
- O nome pode ter o mesmo nome que o contêiner pai.
- A pasta não pode ser renomeada depois de criada.

Criar uma pasta

Você pode criar pastas ao fazer upload de objetos. Para fazer upload de um objeto para uma pasta, você especifica o caminho para a pasta. Se a pasta já existir, o AWS Elemental MediaStore armazenará o objeto na pasta. Se a pasta não existir, o serviço a cria e, em seguida, armazena o objeto na pasta.

Para obter mais informações, consulte [the section called “Fazer upload de um objeto”](#).

Excluir uma pasta

Você pode excluir pastas somente se estiverem vazias, não é possível excluir pastas que contêm objetos.

O AWS Elemental MediaStore exclui automaticamente uma pasta quando você exclui o último objeto nela. O serviço também exclui todas as pastas vazias acima dessa pasta. Por exemplo, suponha que você tenha uma pasta chamada `premium` que não contenha nenhum arquivo, mas contenha uma subpasta denominada `canada`. A subpasta `canada` contém um arquivo chamado `mLaw.ts`. Se você excluir o arquivo `mLaw.ts`, o serviço excluirá as pastas `premium` e `canada`. Essa exclusão automática se aplica apenas a pastas. O serviço não exclui contêineres vazios.

Para obter mais informações, consulte [Excluir um objeto](#).

Objetos no AWS Elemental MediaStore

Os ativos do AWS Elemental MediaStore são chamados de objetos. Você pode fazer upload de um objeto para um contêiner ou para uma pasta no contêiner.

No MediaStore, é possível fazer upload, fazer download e excluir objetos:

- **Upload (Fazer upload):** adicionar um objeto a um contêiner ou uma pasta. Isso não é o mesmo que criar um objeto. É necessário criar seus objetos localmente antes de fazer upload deles no MediaStore.
- **Fazer download:** copiar um objeto do MediaStore para outro local. Isso não remove o objeto do MediaStore.
- **Excluir:** remover completamente um objeto do MediaStore. Você pode excluir objetos individualmente, ou você pode [adicionar uma política de ciclo de vida de objetos](#) para excluir automaticamente objetos dentro de um contêiner após uma duração especificada.

O MediaStore aceita todos os tipos de arquivo.

Tópicos

- [Fazer upload de um objeto](#)
- [Visualizar uma lista de objetos](#)
- [Visualizar os detalhes de um objeto](#)
- [Fazer download de um objeto](#)
- [Excluir objetos](#)

Fazer upload de um objeto

Você pode fazer upload de objetos para um contêiner ou para uma pasta em um contêiner. Para fazer upload de um objeto para uma pasta, você especifica o caminho para a pasta. Se a pasta já existir, o AWS Elemental MediaStore armazenará o objeto lá. Se a pasta não existir, o serviço a cria e, em seguida, armazena o objeto na pasta. Para obter mais informações sobre pastas, consulte [Pastas no AWS Elemental MediaStore](#).

É possível usar o console do MediaStore ou a AWS CLI para fazer upload de objetos.

O MediaStore oferece suporte à transferência de objetos em blocos, o que reduz a latência ao disponibilizar o objeto para download enquanto o upload ainda está sendo feito. Para usar esse recurso, defina a disponibilidade de upload do objeto como `streaming`. Você pode definir o valor desse cabeçalho ao [fazer upload do objeto usando a API](#). Se você não especificar esse cabeçalho em sua solicitação, o MediaStore atribuirá o valor padrão de `standard` para a disponibilidade de upload do objeto.

Os tamanhos de objeto não podem exceder 25 MB para disponibilidade padrão de upload e 10 MB para disponibilidade de upload de streaming.

Note

Os nomes de arquivo de objeto podem conter apenas letras, números, pontos (.), sublinhados (_), tildes (~), hifens (-), sinais de igual (=) e dois pontos (:).

Para fazer upload de um objeto (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner. O painel de detalhes do contêiner é exibido.
3. Selecione Upload object (Fazer upload de objeto).
4. Em Target path (Caminho de destino), digite o caminho das pastas. Por exemplo, `premium/canada`. Se alguma das pastas no caminho especificado ainda não existir, o serviço as criará automaticamente.
5. Na seção Object (Objeto), selecione Browse (Navegar).
6. Navegue até a pasta apropriada e escolha um objeto para fazer upload.
7. Selecione Open (Abrir) e Upload (Fazer upload).

Note

Se um arquivo com o mesmo nome já existir na pasta selecionada, o serviço substituirá o arquivo original pelo arquivo carregado.

Para carregar um objeto (AWS CLI)

- Na AWS CLI, use o comando `put-object`. Também é possível incluir qualquer um dos seguintes parâmetros: `content-type`, `cache-control` (para permitir que o chamador controle o comportamento de cache do objeto) e `path` (para colocar o objeto em uma pasta dentro do contêiner).

Note

Após fazer upload do objeto, você poderá editar o `content-type`, `cache-control` ou `path`.

```
aws mediastore-data put-object --endpoint https://  
aaabbbcccddee.data.mediastore.us-west-2.amazonaws.com --body README.md --path /  
folder_name/README.md --cache-control "max-age=6, public" --content-type binary/  
octet-stream --region us-west-2
```

O exemplo a seguir mostra o valor de retorno:

```
{  
  "ContentSHA256":  
    "74b5fdb517f423ed750ef214c44adfe2be36e37d861eafe9c842cbe1bf387a9d",  
  "StorageClass": "TEMPORAL",  
  "ETag": "af3e4731af032167a106015d1f2fe934e68b32ed1aa297a9e325f5c64979277b"  
}
```

Visualizar uma lista de objetos

Você pode usar o console do AWS Elemental MediaStore para visualizar itens (objetos e pastas) armazenados no nível superior de um contêiner ou em uma pasta. Os itens armazenados em uma subpasta do contêiner ou da pasta atual não serão exibidos. Você pode usar a AWS CLI para visualizar uma lista de objetos e pastas em um contêiner, independentemente de quantas pastas ou subpastas estão no contêiner.

Para visualizar uma lista de objetos em um contêiner específico (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.

2. Na página Containers (Contêineres), selecione o nome do contêiner que tem a pasta que você deseja visualizar.
3. Escolha o nome da pasta na lista.

Uma página de detalhes é exibida, mostrando todas as pastas e objetos que estão armazenados na pasta.

Para visualizar uma lista de objetos em uma pasta específica (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner que tem a pasta que você deseja visualizar.

Uma página de detalhes é exibida, mostrando todas as pastas e objetos que estão armazenados no contêiner.

Para visualizar uma lista de objetos e pastas em um contêiner específico (AWS CLI)

- Na AWS CLI, use o comando `list-items`:

```
aws mediastore-data list-items --endpoint https://  
aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com --region us-west-2
```

O exemplo a seguir mostra o valor de retorno:

```
{  
  "Items": [  
    {  
      "ContentType": "image/jpeg",  
      "LastModified": 1563571859.379,  
      "Name": "filename.jpg",  
      "Type": "OBJECT",  
      "ETag":  
      "543ab21abcd1a234ab123456a1a2b12345ab12abc12a1234abc1a2bc12345a12",  
      "ContentLength": 3784  
    },  
    {  
      "Type": "FOLDER",  
      "Name": "ExampleLiveDemo"  
    }  
  ]  
}
```

```
    }  
  ]  
}
```

Note

Os objetos que estão sujeitos a uma regra `seconds_since_create` não são incluídos em uma resposta `list-items`.

Para visualizar uma lista de objetos e pastas em uma pasta específica (AWS CLI)

- Na AWS CLI, use o comando `list-items`, com o nome da pasta especificada no final da solicitação:

```
aws mediastore-data list-items --endpoint https://  
aaabbbccdddee.data.mediastore.us-west-2.amazonaws.com --path /folder_name --  
region us-west-2
```

O exemplo a seguir mostra o valor de retorno:

```
{  
  "Items": [  
    {  
      "Type": "FOLDER",  
      "Name": "folder_1"  
    },  
    {  
      "LastModified": 1563571940.861,  
      "ContentLength": 2307346,  
      "Name": "file1234.jpg",  
      "ETag":  
"111a1a22222a1a1a222abc333a444444b55ab1111ab2222222222ab333333a2b",  
      "ContentType": "image/jpeg",  
      "Type": "OBJECT"  
    }  
  ]  
}
```

Note

Os objetos que estão sujeitos a uma regra `seconds_since_create` não são incluídos em uma resposta `list-items`.

Visualizar os detalhes de um objeto

Depois de fazer upload de um objeto, o AWS Elemental MediaStore armazena detalhes como data de modificação, tamanho do conteúdo, ETag (tag de entidade) e tipo de conteúdo. Para saber como os metadados de um objeto são usados, consulte [Interação do MediaStore com caches HTTP](#).

Para visualizar os detalhes de um objeto (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner que tem o objeto que você deseja visualizar.
3. Se o objeto do qual você deseja visualizar estiver em uma pasta, continue escolhendo os nomes das pastas até ver o objeto.
4. Escolha o nome do objeto.

Uma página de detalhes será exibida, mostrando informações sobre o objeto.

Para visualizar os detalhes de um objeto (AWS CLI)

- Na AWS CLI, use o comando `describe-object`:

```
aws mediastore-data describe-object --endpoint https://  
aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --path /folder_name/  
file1234.jpg --region us-west-2
```

O exemplo a seguir mostra o valor de retorno:

```
{  
  "ContentType": "image/jpeg",  
  "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",  
  "ContentLength": "2307346",
```


Para fazer download de parte de um objeto (AWS CLI)

- Na AWS CLI, use o comando `get-object` e especifique um intervalo.

```
aws mediastore-data get-object --endpoint https://  
aaabbccdddee.data.mediastore.us-west-2.amazonaws.com --path /folder_name/  
README.md --range="bytes=0-100" README2.md --region us-west-2
```

O exemplo a seguir mostra o valor de retorno:

```
{  
  "StatusCode": 206,  
  "ContentRange": "bytes 0-100/2307346",  
  "ContentLength": "101",  
  "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",  
  "ContentType": "image/jpeg",  
  "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3"  
}
```

Excluir objetos

O AWS Elemental MediaStore oferece diferentes opções para excluir objetos de contêineres:

- [Excluir um objeto individual](#). Não há cobranças aplicáveis.
- [Esvazie um contêiner](#) para excluir todos os objetos dentro de um contêiner de uma vez. Como esse processo usa chamadas de API, cobranças de API normais são aplicáveis.
- [Adicione uma política de ciclo de vida do objeto](#) para excluir objetos quando atingirem uma determinada idade. Não há cobranças aplicáveis.

Excluir um objeto

Você pode excluir objetos individualmente usando o console ou a AWS CLI. Como alternativa, é possível [adicionar uma política de ciclo de vida do objeto](#) para excluir objetos automaticamente depois que eles atingirem uma certa idade em um contêiner, ou é possível [esvaziar um contêiner](#) para excluir todos os objetos dentro dele.

Note

Quando você exclui o único objeto em uma pasta, o AWS Elemental MediaStore exclui automaticamente a pasta e todas as pastas vazias acima dessa pasta. Por exemplo, suponha que você tenha uma pasta chamada `premium` que não contenha nenhum arquivo, mas contenha uma subpasta denominada `canada`. A subpasta `canada` contém um arquivo chamado `m1aw.ts`. Se você excluir o arquivo `m1aw.ts`, o serviço excluirá as pastas `premium` e `canada`.

Para excluir um objeto (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), selecione o nome do contêiner que tem o objeto que você deseja excluir.
3. Se o objeto que você deseja excluir estiver em uma pasta, continue escolhendo os nomes das pastas até ver o objeto.
4. Escolha a opção à esquerda do nome do objeto.
5. Escolha Delete (Excluir).

Para excluir um objeto (AWS CLI)

- Na AWS CLI, use o comando `delete-object`.

Exemplo:

```
aws mediastore-data --region us-west-2 delete-object --endpoint=https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com --path=/folder_name/README.md
```

Este comando não tem valor de retorno.

Esvaziar um contêiner

É possível esvaziar um contêiner para excluir todos os objetos armazenados nele. Como alternativa, é possível adicionar uma [política de ciclo de vida do objeto](#) para excluir objetos automaticamente depois que eles atingem uma certa idade em um contêiner, ou [excluir objetos individualmente](#).

Como esvaziar um contêiner (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), escolha a opção para o contêiner que você deseja esvaziar.
3. Escolha Empty container (Esvaziar contêiner). Uma mensagem de confirmação será exibida.
4. Confirme se deseja esvaziar o contêiner inserindo o nome do contêiner no campo de texto e escolha Esvaziar.

Segurança no AWS Elemental MediaStore

A segurança para com a nuvem na AWS é a nossa maior prioridade. Como cliente da AWS, você se beneficiará de data centers e arquiteturas de rede criados para atender aos requisitos das empresas com as maiores exigências de segurança.

A segurança é uma responsabilidade compartilhada entre a AWS e você. O [modelo de responsabilidade compartilhada](#) descreve isso como a segurança da nuvem e segurança na nuvem:

- **Segurança da nuvem:** AWS é responsável pela proteção da infraestrutura que executa AWS produtos da Nuvem AWS na AWS. A também fornece serviços que podem ser usados com segurança. Auditores de terceiros testam e verificam regularmente a eficácia da nossa segurança como parte dos [AWS Programas de conformidade](#). Para saber mais sobre os programas de conformidade que se aplicam ao AWS Elemental MediaStore, consulte [Serviços da AWS no escopo por programa de conformidade](#).
- **Segurança na nuvem:** sua responsabilidade é determinada pelo serviço da AWS 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 a entender como aplicar o modelo de responsabilidade compartilhada ao usar o MediaStore. Os tópicos a seguir mostram como configurar o MediaStore para atender aos seus objetivos de segurança e conformidade. Saiba também como usar outros serviços da AWS que ajudam a monitorar e proteger os recursos do MediaStore.

Tópicos

- [Proteção de dados no AWS Elemental MediaStore](#)
- [Identity and Access Management para o AWS Elemental MediaStore](#)
- [Logging e monitoramento no AWS Elemental MediaStore](#)
- [Validação de conformidade para o AWS Elemental MediaStore](#)
- [Resiliência no AWS Elemental MediaStore](#)
- [Segurança da infraestrutura no AWS Elemental MediaStore](#)
- [Prevenção contra o ataque do “substituto confuso” em todos os serviços](#)

Proteção de dados no AWS Elemental MediaStore

O AWS [modelo de responsabilidade compartilhada](#) se aplica à proteção de dados no AWS Elemental MediaStore. Conforme descrito nesse modelo, a AWS é responsável por proteger a infraestrutura global que executa toda a Nuvem AWS. Você é responsável por manter o controle sobre seu 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 a privacidade de dados, consulte as [Perguntas frequentes sobre privacidade de dados](#). Para mais informações sobre a proteção de dados na Europa, consulte o artigo [AWS Shared Responsibility Model and GDPR](#) no Blog de segurança da AWS.

Para fins de proteção de dados, recomendamos que você proteja as credenciais da Conta da AWS e configure as contas de usuário individuais com o AWS IAM Identity Center ou o 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 [multi-factor authentication]) com cada conta.
- Use SSL/TLS para se comunicar com os atributos da AWS. Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure o registro em log das atividades da API e do usuário com o .AWS CloudTrail
- Use AWS as soluções de criptografia da , juntamente com todos os controles de segurança padrão dos Serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sigilosos armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-2 ao acessar a AWS por meio de uma interface de linha de comandos ou uma API, use um endpoint do FIPS. Para ter mais informações sobre endpoints do FIPS, consulte [Federal Information Processing Standard \(FIPS\) 140-2](#).

É altamente recomendável que nunca sejam colocadas informações de identificação confidenciais, como endereços de e-mail dos seus clientes, em marcações ou campos de formato livre, como um campo Name (Nome). Isso inclui trabalhar com o MediaStore ou outros Serviços da AWS usando console, API, AWS CLI ou AWS SDKs. Quaisquer dados inseridos em tags ou campos de texto de formato livre usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se

Se você fornecer um URL para um servidor externo, recomendamos fortemente que não sejam incluídas informações de credenciais no URL para validar a solicitação a esse servidor.

Criptografia de dados

O EBS criptografa o volume com uma chave de dados usando o algoritmo AES-256 padrão do setor. Recomendamos que você use o MediaStore para proteger seus dados das seguintes formas:

- Crie uma política de contêiner para controlar os direitos de acesso a todas as pastas e objetos desse contêiner. Para obter mais informações, consulte [the section called “Políticas de contêiner”](#).
- Crie uma política de compartilhamento de recursos entre origens (CORS) para permitir o acesso entre origens de maneira seletiva aos recursos do MediaStore. Com o CORS, você pode permitir que aplicativos web de clientes carregados em um domínio interajam com recursos em outro domínio. Para obter mais informações, consulte [the section called “Políticas de CORS”](#).

Identity and Access Management para o AWS Elemental MediaStore

O AWS Identity and Access Management (IAM) é um AWS service (Serviço da AWS) que ajuda o administrador a controlar o acesso aos recursos da AWS de forma segura. Os administradores do IAM controlam quem pode ser autenticado (fazer login) e autorizado (ter permissões) para usar recursos do MediaStore. O IAM é um AWS service (Serviço da AWS) que pode ser usado sem custo adicional.

Tópicos

- [Público](#)
- [Autenticando com identidades](#)
- [Como gerenciar acesso usando políticas](#)
- [Como o AWS Elemental MediaStore funciona com o IAM](#)
- [Exemplos de políticas baseadas em identidade para o AWS Elemental MediaStore](#)
- [Solução de problemas de identidade e acesso ao AWS Elemental MediaStore](#)

Público

O uso do AWS Identity and Access Management (IAM) varia dependendo do trabalho realizado no MediaStore.

Usuário do serviço: se você usa o serviço MediaStore para fazer o trabalho, o administrador fornece as credenciais e as permissões necessárias. À medida que usar mais atributos do MediaStore para fazer seu trabalho, você poderá precisar de permissões adicionais. Entender como o acesso é gerenciado pode ajudá-lo a solicitar as permissões corretas ao seu administrador. Se não for possível acessar um atributo no MediaStore, consulte [Solução de problemas de identidade e acesso ao AWS Elemental MediaStore](#).

Administrador do serviço: se você for o responsável pelos recursos do MediaStore na empresa, provavelmente terá acesso total ao MediaStore. Cabe a você determinar quais atributos e recursos do MediaStore os usuários do serviço poderão acessar. Assim, você deve enviar solicitações ao administrador do IAM para alterar as permissões dos usuários de seu serviço. Revise as informações nesta página para entender a Introdução ao IAM. Para saber mais sobre como a empresa pode usar o IAM com o MediaStore, consulte [Como o AWS Elemental MediaStore funciona com o IAM](#).

Administrador do IAM: se você for um administrador do IAM, talvez queira saber detalhes sobre como é possível criar políticas para gerenciar o acesso ao MediaStore. Para visualizar exemplos de políticas baseadas em identidade do MediaStore que podem ser usadas no IAM, consulte [Exemplos de políticas baseadas em identidade para o AWS Elemental MediaStore](#).

Autenticando com identidades

A autenticação é a forma como você faz login na AWS usando suas credenciais de identidade. É necessário ser autenticado (fazer login na AWS) como o usuário raiz da Usuário raiz da conta da AWS, como usuário do IAM ou assumindo um perfil do IAM.

Você pode fazer login na AWS como uma identidade federada usando credenciais fornecidas por uma fonte de identidades. Os usuários do AWS IAM Identity Center (IAM Identity Center), a autenticação única da empresa e as suas credenciais do Google ou do Facebook são exemplos de identidades federadas. Quando você faz login como uma identidade federada, o administrador já configurou anteriormente a federação de identidades utilizando perfis do IAM. Quando você acessa a AWS usando a federação, está indiretamente assumindo um perfil.

É possível fazer login no ou no portal de acesso da AWS Management Console dependendo do tipo de usuário que você é. Para obter mais informações sobre como fazer login na AWS, consulte

[How to sign in to your Conta da AWS](#) (Como fazer login na conta da) no Início de Sessão da AWS User Guide (Guia do usuário do).

Se você acessar a AWS programaticamente, a AWS fornecerá um kit de desenvolvimento de software (SDK) e uma interface da linha de comando (CLI) para você assinar criptograficamente as solicitações usando as suas credenciais. Se você não utilizar as ferramentas da AWS, deverá assinar as solicitações por conta própria. Para obter mais informações sobre como usar o método recomendado para assinar solicitações por conta própria, consulte [Assinar AWSsolicitações de API da](#) no Guia do usuário do IAM.

Independentemente do método de autenticação usado, também pode ser exigido que você forneça mais informações de segurança. Por exemplo, a AWS recomenda o uso da autenticação multifator (MFA) para aumentar a segurança de sua conta. Para saber mais, consulte [Autenticação multifator](#) no Guia do usuário do AWS IAM Identity Center e [Usar a autenticação multifator \(MFA\) naAWS](#) no Guia do usuário do IAM.

Usuário raiz da Conta da AWS

Ao criar uma Conta da AWS, você começa com uma identidade de login que tem acesso completo a todos os atributos e Serviços da AWS na conta. Essa identidade, denominada usuário raiz da Conta da AWS, e é acessada por login com o endereço de e-mail e a senha que você usou para criar a conta. É altamente recomendável não utilizar o usuário raiz para tarefas diárias. Proteja as credenciais do usuário raiz e use-as para executar as tarefas que somente ele puder executar. Para obter a lista completa das tarefas que exigem login como usuário raiz, consulte [Tarefas que exigem credenciais de usuário raiz](#) no Guia do usuário do IAM.

Identidade federada

Como prática recomendada, exija que os usuários, inclusive os que precisam de acesso de administrador, usem a federação com um provedor de identidades para acessar Serviços da AWS usando credenciais temporárias.

Identidade federada é um usuário de seu diretório de usuários corporativos, um provedor de identidades da web AWS Directory Service, o , o diretório do Centro de Identidade ou qualquer usuário que acesse os Serviços da AWS usando credenciais fornecidas por meio de uma fonte de identidade. Quando as identidades federadas acessam Contas da AWS, elas assumem perfis que fornecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, recomendamos usar o .AWS IAM Identity Center. Você pode criar usuários e grupos no Centro de Identidade do IAM ou se conectar e sincronizar com

um conjunto de usuários e grupos em sua própria fonte de identidade para uso em todas as suas Contas da AWS e aplicações. Para obter mais informações sobre o Centro de Identidade do IAM, consulte [O que é o Centro de Identidade do IAM?](#) no Guia do usuário do AWS IAM Identity Center.

Usuários e grupos do IAM

Um [usuário do IAM](#) é uma identidade dentro da Conta da AWS que tem permissões específicas para uma única pessoa ou aplicação. Sempre que possível, recomendamos contar com credenciais temporárias em vez de criar usuários do IAM com credenciais de longo prazo, como senhas e chaves de acesso. No entanto, se você tiver casos de utilização específicos que exijam credenciais de longo prazo com usuários do IAM, recomendamos alternar as chaves de acesso. Para obter mais informações, consulte [Altere as chaves de acesso regularmente para casos de uso que exijam credenciais](#) de longo prazo no Guia do usuário do IAM.

Um [grupo do IAM](#) é uma identidade que especifica uma coleção de usuários do IAM. Não é possível fazer login como um grupo. É possível utilizar grupos para especificar permissões para vários usuários de uma vez. Os grupos facilitam o gerenciamento de permissões para grandes conjuntos de usuários. Por exemplo, você pode ter um grupo chamado IAMAdmins e atribuir a esse grupo permissões para administrar atributos do IAM.

Usuários são diferentes de perfis. Um usuário é exclusivamente associado a uma pessoa ou a uma aplicação, mas um perfil pode ser assumido por qualquer pessoa que precisar dele. Os usuários têm credenciais permanentes de longo prazo, mas os perfis fornecem credenciais temporárias. Para saber mais, consulte [Quando criar um usuário do IAM \(em vez de uma função\)](#) no Guia do usuário do IAM.

Perfis do IAM

Um [perfil do IAM](#) é uma identidade dentro da Conta da AWS que tem permissões específicas. Ele é semelhante a um usuário do IAM, mas não está associado a uma pessoa específica. É possível assumir temporariamente um perfil do IAM no AWS Management Console [alternando perfis](#). É possível assumir um perfil chamando uma operação de API da AWS CLI ou da AWS, ou usando um URL personalizado. Para obter mais informações sobre métodos para o uso de perfis, consulte [Uso de funções do IAM](#) no Guia do usuário do IAM.

Perfis do IAM com credenciais temporárias são úteis nas seguintes situações:

- **Acesso de usuário federado:** para atribuir permissões a identidades federadas, você pode criar um perfil e definir permissões para ele. Quando uma identidade federada é autenticada, essa identidade é associada ao perfil e recebe as permissões definidas pelo mesmo. Para obter

mais informações sobre perfis para federação, consulte [Criar uma função para um provedor de identidade de terceiros](#) no Guia do usuário do IAM. Se você usar o IAM Identity Center, deverá configurar um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o Centro de identidade do IAM correlaciona o conjunto de permissões a um perfil no IAM. Para obter informações sobre conjuntos de permissões, consulte [Conjuntos de permissões](#) no Guia do usuário do AWS IAM Identity Center.

- Permissões temporárias para usuários do IAM: um usuário ou um perfil do IAM pode assumir um perfil do IAM para obter temporariamente permissões diferentes para uma tarefa específica.
- Acesso entre contas: é possível usar um perfil do IAM para permitir que alguém (uma entidade principal confiável) em outra conta acesse recursos em sua conta. Os perfis são a principal forma de conceder acesso entre contas. No entanto, alguns Serviços da AWS permitem que você anexe uma política diretamente a um atributo (em vez de usar um perfil como proxy). Para saber a diferença entre perfis e políticas baseadas em atributo para acesso entre contas, consulte [Como os perfis do IAM diferem das políticas baseadas em atributo](#) no Guia do usuário do IAM.
- Acesso entre serviços: alguns Serviços da AWS usam atributos em outros Serviços da AWS. Por exemplo, quando você faz uma chamada em um serviço, é comum que esse serviço execute aplicações no Amazon EC2 ou armazene objetos no Amazon S3. Um serviço pode fazer isso usando as permissões do principal de chamada, usando um perfil de serviço ou um perfil vinculado ao serviço.
- Encaminhamento de sessões de acesso (FAS): qualquer pessoa que utilizar uma função ou usuário do IAM para realizar ações na AWS é considerada uma entidade principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O recurso FAS utiliza as permissões da entidade principal que chama um AWS service (Serviço da AWS), combinadas às permissões do AWS service (Serviço da AWS) solicitante, para realizar solicitações para serviços downstream. As solicitações de FAS só são feitas quando um serviço recebe uma solicitação que exige interações com outros Serviços da AWS ou com recursos para serem concluídas. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte [Encaminhar sessões de acesso](#).
- Perfil de serviço: um perfil de serviço é um [perfil do IAM](#) que um serviço assume para realizar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte [Criar um perfil para delegar permissões a um AWS service \(Serviço da AWS\)](#) no Guia do usuário do IAM.
- Função vinculada ao serviço: uma função vinculada a serviço é um tipo de função de serviço vinculada a um AWS service (Serviço da AWS). O serviço pode assumir o perfil para executar

uma ação em seu nome. Os perfis vinculados ao serviço aparecem em sua Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode exibir, mas não pode editar as permissões para perfis vinculados ao serviço.

- Aplicações em execução no Amazon EC2: é possível usar um perfil do IAM para gerenciar credenciais temporárias para aplicações em execução em uma instância do EC2 e fazer solicitações da AWS CLI ou da AWS API. É preferível fazer isso e armazenar chaves de acesso na instância do EC2. Para atribuir um perfil da AWS a uma instância do EC2 e disponibilizá-la para todas as suas aplicações, crie um perfil de instância que esteja anexado a ela. Um perfil de instância contém a perfil e permite que os programas em execução na instância do EC2 obtenham credenciais temporárias. Para mais informações, consulte [Usar um perfil do IAM para conceder permissões a aplicações em execução nas instâncias do Amazon EC2](#) no Guia do usuário do IAM.

Para saber se deseja usar os perfis do IAM, consulte [Quando criar um perfil do IAM \(em vez de um usuário\)](#) no Guia do usuário do IAM.

Como gerenciar acesso usando políticas

Você controla o acesso na AWS criando políticas e anexando-as a identidades ou atributos da AWS. Uma política é um objeto na AWS que, quando associado a uma identidade ou atributo, define suas permissões. A AWS avalia essas políticas quando uma entidade principal (usuário, usuário raiz ou sessão de perfil) faz uma solicitação. As permissões nas políticas determinam se a solicitação será permitida ou negada. A maioria das políticas são armazenadas na AWS como documentos JSON. Para obter mais informações sobre a estrutura e o conteúdo de documentos de políticas JSON, consulte [Visão geral das políticas JSON](#) no Guia do usuário do IAM.

Os administradores podem usar AWS as políticas JSON da para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

Por padrão, usuários e funções não têm permissões. Para conceder aos usuários permissão para executar ações nos recursos de que eles precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM a perfis, e os usuários podem assumir os perfis.

As políticas do IAM definem permissões para uma ação independente do método usado para executar a operação. Por exemplo, suponha que você tenha uma política que permite a ação `iam:GetRole`. Um usuário com essa política pode obter informações de perfil do AWS Management Console, da AWS CLI ou da AWS API.

Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário, grupo de usuários ou perfil do IAM. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Criar políticas do IAM](#) no Guia do usuário do IAM.

As políticas baseadas em identidade podem ser categorizadas ainda mais como políticas embutidas ou políticas gerenciadas. As políticas embutidas são anexadas diretamente a um único usuário, grupo ou função. As políticas gerenciadas são políticas independentes que podem ser anexadas a vários usuários, grupos e perfis na Conta da AWS. As políticas gerenciadas incluem políticas gerenciadas pela AWS e políticas gerenciadas pelo cliente. Para saber como escolher entre uma política gerenciada ou uma política em linha, consulte [Escolher entre políticas gerenciadas e políticas em linha](#) no Guia do usuário do IAM.

Políticas baseadas em recurso

Políticas baseadas em atributos são documentos de políticas JSON que você anexa a um atributo. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recurso, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações uma entidade principal especificado pode executar nesse recurso e em que condições. Você precisa [especificar uma entidade principal](#) em uma política baseada em recurso. As entidades principais podem incluir contas, usuários, perfis, usuários federados ou Serviços da AWS.

Políticas baseadas em atributos são políticas em linha que estão localizadas nesse serviço. Não é possível usar as políticas gerenciadas da AWS do IAM em uma política baseada em atributos.

Listas de controle de acesso (ACLs)

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou funções da conta) têm permissões para acessar um recurso. As ACLs são semelhantes às políticas baseadas em recurso, embora não usem o formato de documento de política JSON.

Amazon S3, AWS WAF e Amazon VPC são exemplos de serviços compatíveis com ACLs. Para saber mais sobre ACLs, consulte [Visão geral da lista de controle de acesso \(ACL\)](#) no Guia do desenvolvedor do Amazon Simple Storage Service.

Outros tipos de política

A AWS aceita tipos de política menos comuns. Esses tipos de política podem definir o máximo de permissões concedidas a você pelos tipos de política mais comuns.

- **Limites de permissões:** um limite de permissões é um atributo avançado no qual você define o máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM (usuário ou perfil do IAM). É possível definir um limite de permissões para uma entidade. As permissões resultantes são a interseção das políticas baseadas em identidade de uma entidade com seus limites de permissões. As políticas baseadas em recurso que especificam o usuário ou perfil no campo `Principal` não são limitadas pelo limite de permissões. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações sobre limites de permissões, consulte [Limites de permissões para identidades do IAM](#) no Guia do usuário do IAM.
- **Políticas de controle de serviço (SCPs):** SCPs são políticas JSON que especificam as permissões máximas para uma organização ou unidade organizacional (UO) no AWS Organizations. O AWS Organizations é um serviço para agrupar e gerenciar centralmente várias Contas da AWS pertencentes à sua empresa. Se você habilitar todos os atributos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as contas. O SCP limita as permissões para entidades em contas-membro, incluindo cada `.Usuário raiz` da conta da AWS. Para obter mais informações sobre as Organizações e SCPs, consulte [How SCPs work \(Como os SCPs funcionam\)](#) no AWS Organizations Guia do usuário do .
- **Políticas de sessão:** são políticas avançadas que você transmite como um parâmetro quando cria de forma programática uma sessão temporária para um perfil ou um usuário federado. As permissões da sessão resultante são a interseção das políticas baseadas em identidade do usuário ou do perfil e das políticas de sessão. As permissões também podem ser provenientes de uma política baseada em atributo. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações, 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 a AWS determina se deve permitir uma solicitação quando há vários tipos de política envolvidos, consulte [Lógica da avaliação](#) de políticas no Guia do usuário do IAM.

Como o AWS Elemental MediaStore funciona com o IAM

Antes de usar o IAM para gerenciar o acesso ao MediaStore, saiba quais atributos do IAM estão disponíveis para uso com o MediaStore.

Atributos do IAM que você pode usar com o AWS Elemental MediaStore

Atributo do IAM	Suporte do MediaStore
Políticas baseadas em identidade	Sim
Políticas baseadas em atributos	Sim
Ações de políticas	Sim
atributos de políticas	Sim
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	Sim
Perfis vinculados ao serviço	Não

Para obter uma visão geral de como o MediaStore e outros serviços da AWS funcionam com a maioria dos atributos do IAM, consulte [Serviços da AWS que funcionam com o IAM](#) no Guia do usuário do IAM.

Políticas do baseadas em identidade para o MediaStore

Suporta políticas baseadas em identidade	Sim
--	-----

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário do IAM, grupo de usuários ou perfil. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Criar políticas do IAM](#) no Guia do usuário do IAM.

Com as políticas baseadas em identidade do IAM, é possível especificar ações ou atributos permitidos ou negados, bem como as condições sob as quais as ações são permitidas ou negadas. Você não pode especificar a entidade principal em uma política baseada em identidade porque ela se aplica ao usuário ou função à qual ela está anexada. Para saber mais sobre todos os elementos que podem ser usados em uma política JSON, consulte [Referência de elementos da política JSON do IAM](#) no Guia do Usuário do IAM.

Exemplos de políticas baseadas em identidade para o MediaStore

Para visualizar exemplos de políticas baseadas em identidade do MediaStore, consulte [Exemplos de políticas baseadas em identidade para o AWS Elemental MediaStore](#).

Políticas baseadas em recursos no MediaStore

Oferece suporte a políticas baseadas em atributos	Sim
---	-----

Políticas baseadas em atributos são documentos de políticas JSON que você anexa a um atributo. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recurso, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações uma entidade principal especificado pode executar nesse recurso e em que condições. Você precisa [especificar uma entidade principal](#) em uma política baseada em recurso. As entidades principais podem incluir contas, usuários, perfis, usuários federados ou Serviços da AWS.

Para permitir o acesso entre contas, você pode especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em atributo. Adicionar uma entidade principal entre contas à política baseada em recurso é apenas metade da tarefa de estabelecimento da relação de confiança. Quando a entidade principal e o atributo estão em diferentes Contas da AWS, um administrador do IAM da conta confiável também deve conceder à

entidade principal (usuário ou perfil) permissão para acessar o atributo. Eles concedem permissão ao anexar uma política baseada em identidade para a entidade. No entanto, se uma política baseada em recurso conceder acesso a uma entidade principal na mesma conta, nenhuma outra política baseada em identidade será necessária. Para obter mais informações, consulte [Como as funções do IAM diferem de políticas baseadas em recurso](#) no Guia do usuário do IAM.

Note

O MediaStore também é compatível com políticas de contêiner que definem que entidades principais (contas, usuários, funções e usuários federados) podem realizar ações no contêiner. Para obter mais informações, consulte [Políticas de contêiner](#).

Ações da política para o MediaStore

Oferece suporte a ações de políticas	Sim
--------------------------------------	-----

Os administradores podem usar as políticas de JSON da AWS para especificar quem tem acesso a 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 você pode usar para permitir ou negar acesso em uma política. As ações de política geralmente têm o mesmo nome que a operação de API da AWS associada. Existem algumas exceções, como ações somente de permissão, que não têm uma operação de API correspondente. Algumas operações também exigem várias ações em uma política. Essas ações adicionais são chamadas de ações dependentes.

Incluem ações em uma política para conceder permissões para executar a operação associada.

Para ver uma lista das ações do MediaStore, consulte [Ações definidas pelo AWS Elemental MediaStore](#) na Referência de autorização do serviço.

As ações de políticas no MediaStore usam o seguinte prefixo antes da ação:

```
mediastore
```

Para especificar várias ações em uma única instrução, separe-as com vírgulas.

```
"Action": [  
  "mediastore:action1",  
  "mediastore:action2"  
]
```

Para visualizar exemplos de políticas baseadas em identidade do MediaStore, consulte [Exemplos de políticas baseadas em identidade para o AWS Elemental MediaStore](#).

Recursos da política para o MediaStore

Oferece suporte a atributos de políticas	Sim
--	-----

Os administradores podem usar as políticas de JSON da AWS para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento de política Resource JSON especifica o objeto ou os objetos aos quais a ação se aplica. As instruções devem incluir um elemento Resource ou um elemento NotResource. Como prática recomendada, especifique um atributo usando seu [Nome do atributo da Amazon \(ARN\)](#). Isso pode ser feito para ações que oferecem suporte a um tipo de atributo específico, conhecido como permissões em nível de atributo.

Para ações que não oferecem suporte a permissões em nível de recurso, como operações de listagem, use um caractere curinga (*) para indicar que a instrução se aplica a todos os recursos.

```
"Resource": "*"
```

Para ver uma lista dos tipos de recursos do MediaStore e seus ARNs, consulte [Recursos definidos pelo AWS Elemental MediaStore](#) na Referência de autorização do serviço. Para saber com quais ações é possível especificar o ARN de cada recurso, consulte [Ações definidas pelo AWS Elemental MediaStore](#).

O recurso de contêiner do MediaStore tem o seguinte ARN:

```
arn:${Partition}:mediastore:${Region}:${Account}:container/${containerName}
```

Para obter mais informações sobre o formato de ARNs, consulte [Nomes de recursos da Amazon \(ARNs\)AWS e namespaces de serviços da](#)

Por exemplo, para especificar o contêiner AwardsShow em sua instrução, use o seguinte ARN:

```
"Resource": "arn:aws:mediastore:us-east-1:111122223333:container/AwardsShow"
```

Chaves de condição da política para o MediaStore

Suporta chaves de condição de política específicas de serviço	Sim
---	-----

Os administradores podem usar as políticas de JSON da AWS para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento Condition (ou Condition bloco de) permite que você especifique condições nas quais uma instrução está em vigor. O elemento Condition é opcional. É possível criar expressões condicionais que usam [agentes de condição](#), como “igual a” ou “menor que”, para fazer a condição da política corresponder aos valores na solicitação.

Se você especificar vários elementos Condition em uma instrução ou várias chaves em um único Condition elemento, a AWS os avaliará usando uma operação lógica AND. Se você especificar vários valores para uma única chave de condição, a AWS avaliará a condição usando uma operação lógica OR. Todas as condições devem ser atendidas antes que as permissões da instrução sejam concedidas.

Você também pode utilizar variáveis de espaço reservado ao especificar as condições. Por exemplo, é possível conceder a um usuário do IAM permissão para acessar um atributo somente se ele estiver marcado com seu nome de usuário do IAM. Para obter mais informações, consulte [Elementos de política do IAM: variáveis e tags](#) no Guia do usuário do IAM.

A AWS oferece suporte a chaves de condição globais e chaves de condição específicas do serviço. Para ver todas as AWS chaves de condição globais da , consulte [AWSChaves de contexto de condição globais da](#) no Guia do usuário do IAM.

Para ver uma lista de chaves de condição do MediaStore, consulte [Chaves de condição do AWS Elemental MediaStore](#) na Referência de autorização do serviço. Para saber com quais ações e

recursos é possível usar uma chave de condição, consulte [Ações definidas pelo AWS Elemental MediaStore](#).

Para visualizar exemplos de políticas baseadas em identidade do MediaStore, consulte [Exemplos de políticas baseadas em identidade para o AWS Elemental MediaStore](#).

ACLs no MediaStore

Oferece suporte a 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 recurso, embora não usem o formato de documento de política JSON.

ABAC com MediaStore

Oferece suporte a ABAC (tags em políticas)	Parcial
--	---------

O controle de acesso baseado em atributo (ABAC) é uma estratégia de autorização que define permissões com base em atributos. Na AWS, esses atributos são chamados de tags. É possível anexar tags a entidades do IAM (usuários ou perfis) e a muitos atributos da AWS. A marcação de entidades e atributos é a primeira etapa do ABAC. Em seguida, você cria políticas de ABAC para permitir operações quando a tag da entidade principal corresponder à tag do atributo que ela está tentando acessar.

O ABAC é útil em ambientes que estão crescendo rapidamente e ajuda em situações onde o gerenciamento de políticas se torna um problema.

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 `aws:TagKeys` chaves de condição.

Se um serviço oferecer suporte às três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço oferecer suporte às três chaves de condição somente para alguns tipos de recursos, o valor será Parcial.

Para obter mais informações sobre o ABAC, consulte [O que é ABAC?](#) no Guia do usuário do IAM. Para visualizar um tutorial com etapas para configurar o ABAC, consulte [Usar controle de acesso baseado em atributos \(ABAC\)](#) no Guia do usuário do IAM.

Uso de credenciais temporárias com o MediaStore

Oferece suporte a credenciais temporárias	Sim
---	-----

Alguns Serviços da AWS não funcionam quando você faz login usando credenciais temporárias. Para obter informações adicionais, incluindo quais Serviços da AWS funcionam com credenciais temporárias, consulte [Serviços da AWS que funcionam com o IAM](#) no Guia do usuário do IAM.

Você está usando credenciais temporárias se faz login no AWS Management Console usando qualquer método, exceto um nome de usuário e uma senha. Por exemplo, quando você acessa a AWS usando o link de autenticação única (SSO) da sua empresa, esse processo cria automaticamente credenciais temporárias. Você também cria automaticamente credenciais temporárias quando faz login no console como usuário e, em seguida, alterna perfis. Para obter mais informações sobre como alternar perfis, consulte [Alternar para um perfil \(console\)](#) no Guia do usuário do IAM.

Você pode criar credenciais temporárias manualmente usando a AWS CLI ou a API da AWS. Em seguida, você pode usar essas credenciais temporárias para acessar a AWS. A AWS recomenda que você gere credenciais temporárias dinamicamente em vez de usar chaves de acesso de longo prazo. Para obter mais informações, consulte [Credenciais de segurança temporárias no IAM](#).

Permissões de entidade principal entre serviços para o MediaStore

Suporte para o recurso Encaminhamento de sessões de acesso (FAS)	Sim
--	-----

Quando você usa um usuário ou perfil do IAM para executar ações na AWS, você é considerado uma entidade principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O recurso FAS utiliza as permissões da entidade principal que chama um AWS service (Serviço da AWS), combinadas às permissões do AWS service (Serviço da AWS) solicitante, para realizar solicitações para serviços downstream. As solicitações de FAS só são feitas

quando um serviço recebe uma solicitação que exige interações com outros Serviços da AWS ou com recursos para serem concluídas. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte [Encaminhar sessões de acesso](#).

Funções de serviço para o MediaStore

Oferece suporte a perfis de serviço	Sim
-------------------------------------	-----

O perfil de serviço é um [perfil do IAM](#) que um serviço assume para realizar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte [Criar um perfil para delegar permissões a um AWS service \(Serviço da AWS\)](#) no Guia do usuário do IAM.

Warning

Alterar as permissões de um perfil de serviço pode prejudicar a funcionalidade do MediaStore. Edite os perfis de serviço somente quando o MediaStore orientar você a fazê-lo.

Funções vinculadas ao serviço do MediaStore

Oferece suporte a perfis vinculados ao serviço	Não
--	-----

Uma função vinculada ao serviço é um tipo de perfil de serviço vinculada a um AWS service (Serviço da AWS). O serviço pode presumir a função de executar uma ação em seu nome. Os perfis vinculados ao serviço aparecem em sua Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não pode editar as permissões para perfis vinculados ao serviço.

Para obter detalhes sobre como criar ou gerenciar funções vinculadas a serviços, consulte [Serviços do AWS que funcionam com o IAM](#). Encontre um serviço na tabela que inclua um Yes na coluna Perfil vinculado ao serviço. Escolha o link Sim para visualizar a documentação do perfil vinculado a esse serviço .

Exemplos de políticas baseadas em identidade para o AWS Elemental MediaStore

Por padrão, usuários e funções não têm permissão para criar ou modificar recursos do MediaStore. Eles também não podem executar tarefas usando o AWS Management Console, a AWS Command Line Interface (AWS CLI) ou a API AWS. Para conceder aos usuários permissão para executar ações nos recursos de que precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM aos perfis, e os usuários podem assumir os perfis.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documento de política JSON, consulte [Criação de políticas do IAM](#) no Guia do Usuário do IAM.

Para obter detalhes sobre ações e tipos de recurso definidos pelo MediaStore, incluindo o formato dos ARNs para cada tipo de recurso, consulte [Ações, recursos e chaves de condição do AWS Elemental MediaStore](#) na Referência de autorização do serviço.

Tópicos

- [Melhores práticas de políticas](#)
- [Uso do console do MediaStore](#)
- [Permitir que usuários visualizem suas próprias permissões](#)

Melhores práticas de políticas

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir recursos do MediaStore em sua conta. Essas ações podem incorrer em custos para a Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com AWS as políticas gerenciadas pela e avance para as permissões de privilégio mínimo: para começar a conceder permissões a seus usuários e workloads, use as AWS políticas gerenciadas pela que concedem permissões para muitos casos de uso comuns. Elas estão disponíveis na sua Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo cliente da AWS específicas para seus casos de uso. Para obter mais informações, consulte [Políticas gerenciadas pela AWS](#) ou [Políticas gerenciadas pela AWS para perfis 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 atributos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para obter mais informações 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: você pode adicionar uma condição às políticas para limitar o acesso a ações e atributos. Por exemplo, você pode escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso a ações de serviço, se elas forem usadas por meio de um AWS service (Serviço da AWS) específico, como o AWS CloudFormation. Para obter mais informações, consulte [Elementos de política JSON do IAM: Condition](#) 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 100 verificações de política e recomendações acionáveis para ajudá-lo a criar políticas seguras e funcionais. Para obter mais informações, consulte [Validação de políticas do IAM Access Analyzer](#) no Guia do usuário do IAM.
 - Exigir autenticação multifator (MFA): se houver um cenário que exija usuários do IAM ou um usuário raiz em sua Conta da AWS, ative a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para obter mais informações, consulte [Configuração de acesso à API protegido por MFA](#) no Guia do usuário do IAM.

Para obter mais informações sobre as práticas recomendadas do IAM, consulte [Práticas recomendadas de segurança no IAM](#) no Guia do usuário do IAM.

Uso do console do MediaStore

Para acessar o console do AWS Elemental MediaStore, você deve ter um conjunto mínimo de permissões. Essas permissões devem autorizar você a listar e visualizar detalhes sobre os recursos do MediaStore na sua Conta da AWS. Se você criar uma política baseada em identidade que seja mais restritiva do que as permissões mínimas necessárias, o console não funcionará como pretendido para entidades (usuários ou perfis) com essa política.

Não é necessário conceder permissões mínimas do console para usuários que fazem chamadas somente à AWS CLI ou à AWS API. Em vez disso, permita o acesso somente a ações que correspondam a operação de API que estiverem tentando executar.

Para garantir que os usuários e os perfis ainda possam usar o console do MediaStore, anexe também o *ConsoleAccess* do MediaStore ou a política gerenciada *ReadOnly* AWS às entidades. Para obter mais informações, consulte [Adicionando Permissões a um Usuário](#) no Guia do Usuário do IAM.

Permitir que usuários visualizem suas próprias permissões

Este exemplo mostra como é possível criar uma política que permite 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 de forma programática usando a AWS CLI ou a AWS API.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

```
]
}
```

Solução de problemas de identidade e acesso ao AWS Elemental MediaStore

Use as seguintes informações para ajudar a diagnosticar e corrigir problemas comuns que podem ser encontrados ao trabalhar com o MediaStore e o IAM.

Tópicos

- [Não tenho autorização para executar uma ação no MediaStore](#)
- [Não estou autorizado a executar iam:PassRole](#)
- [Quero permitir que as pessoas fora da minha Conta da AWS acessem meus recursos do MediaStore](#)

Não tenho autorização para executar uma ação no MediaStore

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 `mediastore:GetWidget` fictícias.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
mediastore:GetWidget on resource: my-example-widget
```

Nesse caso, a política do usuário mateojackson deve ser atualizada para permitir o acesso ao atributo *my-example-widget* usando a ação `mediastore:GetWidget`.

Se você precisar de ajuda, entre em contato com seu administrador AWS. Seu administrador é a pessoa que forneceu suas credenciais de login.

Não estou autorizado a executar iam:PassRole

Se você receber uma mensagem de erro informando que não tem autorização para executar a ação `iam:PassRole`, as suas políticas deverão ser atualizadas para permitir a passagem de um perfil para o MediaStore.

Alguns Serviços da AWS permitem que você passe um perfil existente para o serviço, em vez de criar um novo perfil de serviço ou perfil vinculado ao serviço. Para fazer isso, é preciso ter permissões para passar o perfil para o serviço.

O erro de exemplo a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta usar o console para executar uma ação no MediaStore. 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 você precisar de ajuda, entre em contato com seu administrador AWS. Seu administrador é a pessoa que forneceu suas credenciais de login.

Quero permitir que as pessoas fora da minha Conta da AWS acessem meus recursos do MediaStore

Você pode criar um perfil que os usuários de outras contas ou pessoas fora da sua organização possam usar para acessar seus recursos. Você pode especificar quem é confiável para assumir o perfil. Para serviços que oferecem suporte a políticas baseadas em recurso ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se o MediaStore oferece suporte a esses atributos, consulte [Como o AWS Elemental MediaStore funciona com o IAM](#).
- Para saber como conceder acesso a seus atributos em todas as Contas da AWS pertencentes a você, consulte [Fornecimento de acesso a um usuário do IAM em outra Conta da AWS pertencente a você](#) no Guia de usuário do IAM.

- Para saber como conceder acesso a seus recursos para Contas da AWS terceirizadas, consulte [Fornecimento de acesso a Contas da AWS pertencentes 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 usar perfis e políticas baseadas em recurso para acesso entre contas, consulte [Como as funções do IAM diferem de políticas baseadas em recurso](#) no Guia do usuário do IAM.

Logging e monitoramento no AWS Elemental MediaStore

Esta seção fornece uma visão geral das opções de registro em log e monitoramento no AWS Elemental MediaStore para fins de segurança. Para obter mais informações sobre o registro e o monitoramento no MediaStore, consulte [Monitoramento e marcação no AWS Elemental MediaStore](#).

O monitoramento é uma parte importante da manutenção da confiabilidade, da disponibilidade e do desempenho do AWS Elemental MediaStore e de soluções da AWS. Você deve coletar dados de monitoramento de todas as partes de sua solução da AWS para depurar uma falha de vários pontos com facilidade, caso ocorra. A AWS fornece várias ferramentas para monitorar seus recursos do MediaStore e responder a incidentes em potencial.

Alarmes do Amazon CloudWatch

Ao usar alarmes do CloudWatch, você observa uma única métrica durante um período especificado. Se a métrica exceder determinado limite, uma notificação será enviada para um tópico do Amazon SNS ou para uma política do AWS Auto Scaling. Os alarmes do CloudWatch não invocam ações só porque estão em um determinado estado. O estado deve ter sido alterado e mantido por uma quantidade especificada de períodos. Para obter mais informações, consulte [Monitoramento com CloudWatch](#).

Logs do AWS CloudTrail

O CloudTrail fornece um registro de ações executadas por um usuário, um perfil ou um serviço da AWS no AWS Elemental MediaStore. Usando as informações coletadas pelo CloudTrail, é possível determinar a solicitação feita para o MediaStore, o endereço IP no qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita, além de detalhes adicionais. Para obter mais informações, consulte [Registrar em log chamadas de API com o CloudTrail](#).

AWS Trusted Advisor

O Trusted Advisor AWS conta com as melhores práticas aprendidas com o atendimento a centenas de milhões de clientes da Trusted Advisor. O inspeciona seu ambiente da e faz recomendações quando há oportunidades para economizar dinheiro, melhorar o desempenho e a disponibilidade do sistema e ajuda a corrigir falhas 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](#).

Validação de conformidade para o AWS Elemental MediaStore

Para saber se um AWS service (Serviço da AWS) está no escopo de programas de conformidade específicos, consulte [Serviços da AWS no escopo por programa de conformidade](#) e selecione o programa de conformidade em que você está interessado. Para obter informações gerais, consulte [AWS Programas de conformidade](#).

É possível fazer download de relatórios de auditoria de terceiros usando o AWS Artifact. Para obter mais informações, consulte [Downloading Reports in AWS Artifact](#).

Sua responsabilidade de conformidade ao usar o Serviços da AWS é determinada pela confidencialidade dos seus dados, pelos objetivos de conformidade da sua empresa e pelos regulamentos e leis aplicáveis. A AWS fornece os seguintes atributos para ajudar com a conformidade:

- [Guias de referência rápida de conformidade e segurança](#) - estes guias de implantação discutem considerações sobre arquitetura e fornecem as etapas para a implantação de ambientes de linha de base focados em segurança e conformidade na AWS.
- [Architecting for HIPAA Security and Compliance on Amazon Web Services \(Arquitetura para segurança e conformidade com HIPAA no Amazon Web Services\)](#): esse estudo técnico descreve como as empresas podem usar a AWS para criar aplicações adequadas aos padrões HIPAA.

Note

Nem todos os Serviços da AWS estão qualificados pela HIPAA. Para obter mais informações, consulte a [Referência dos serviços qualificados pela HIPAA](#).

- [atributos de conformidade da AWS](#): essa coleção de manuais e guias pode ser aplicada a seu setor e local.
- [Guias de conformidade do cliente da AWS](#): entenda o modelo de responsabilidade compartilhada sob a ótica da conformidade. Os guias resumem as práticas recomendadas para proteção de Serviços da AWS e mapeiam as diretrizes para controles de segurança em várias estruturas (incluindo o Instituto Nacional de Padrões e Tecnologia (NIST), o Conselho de Padrões de Segurança do Setor de Cartões de Pagamento (PCI) e a Organização Internacional de Padronização (ISO)).
- [Avaliar atributos com regras](#) no AWS Config Guia do desenvolvedor: o serviço AWS Config avalia como as configurações de atributos estão em conformidade com práticas internas, diretrizes do setor e regulamentos.
- [AWS Security Hub](#): este AWS service (Serviço da AWS) fornece uma visão abrangente do seu estado de segurança na AWS. O Security Hub usa controles de segurança para avaliar os atributos da AWS e verificar a conformidade com os padrões e as práticas recomendadas do setor de segurança. Para obter uma lista dos serviços e controles aceitos, consulte a [Referência de controles do Security Hub](#).
- [AWS Audit Manager](#): esse AWS service (Serviço da AWS) ajuda a auditar continuamente seu uso da para simplificar a forma como você gerencia os riscos e a conformidade com regulamentos e padrões do setor.

Resiliência no AWS Elemental MediaStore

A infraestrutura global da AWS se baseia em Regiões da AWS e zonas de disponibilidade. A Regiões da AWS oferece várias zonas de disponibilidade separadas e isoladas fisicamente que são conectadas com baixa latência, throughputs elevadas e em redes altamente redundantes. Com as zonas de disponibilidade, é possível projetar e operar aplicativos e bancos de dados que automaticamente executam o failover entre as zonas sem interrupção. As zonas de disponibilidade são mais altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre Regiões da AWS e zonas de disponibilidade, consulte [AWS Infraestrutura global](#).

Além da infraestrutura global da AWS, o MediaStore oferece vários atributos para ajudar a oferecer suporte às suas necessidades de resiliência de dados e backup.

Segurança da infraestrutura no AWS Elemental MediaStore

Por ser um serviço gerenciado, o AWS Elemental MediaStore é protegido pela segurança da rede global da AWS. Para obter informações sobre serviços de segurança da AWS e como a AWS protege a infraestrutura, consulte [Segurança na nuvem da AWS](#). Para projetar seu ambiente da AWS usando as práticas recomendadas de segurança de infraestrutura, consulte [Proteção de infraestrutura](#) em Pilar de segurança: AWS Well-Architected Framework.

Você usa chamadas de API publicadas pela AWS para acessar o MediaStore por meio da rede. Os clientes precisam oferecer suporte para:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Conjuntos de criptografia com sigilo de encaminhamento perfeito (perfect forward secrecy, ou PFS) como DHE (Ephemeral Diffie-Hellman, ou Efêmero Diffie-Hellman) ou ECDHE (Ephemeral Elliptic Curve Diffie-Hellman, ou Curva elíptica efêmera Diffie-Hellman). A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

Além disso, as solicitações devem ser assinadas utilizando um ID da chave de acesso e uma chave de acesso secreta associada a uma entidade principal do IAM. Ou você pode usar o [AWS Security Token Service](#) (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

Prevenção contra o ataque do “substituto confuso” em todos os serviços

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 executá-la. Em AWS, a personificação entre serviços pode resultar no problema do “substituto confuso”. A imitação entre serviços pode ocorrer quando um serviço (o serviço de chamada) chama outro serviço (o serviço chamado). O serviço de chamada pode ser manipulado para utilizar as suas permissões para atuar nos recursos de outro cliente em que, de outra forma, ele não teria permissão para acessar. Para evitar isso, o 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 o uso das chaves de contexto de condição global [aws:SourceArn](#) e [aws:SourceAccount](#) em políticas de recursos para limitar as permissões que o AWS Elemental MediaStore concede a outro serviço no recurso para o recurso. Use `aws:SourceArn` se quiser que

apenas um recurso seja associado ao acesso entre serviços. Use `aws:SourceAccount` se quiser permitir que qualquer recurso nessa conta seja associado ao uso entre serviços.

A maneira mais eficaz de se proteger contra o problema do substituto confuso 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 estiver especificando vários recursos, use a chave de condição de contexto global `aws:SourceArn` com caracteres curingas (*) para as partes desconhecidas do ARN. Por exemplo, `arn:aws:service:*:123456789012:*`.

Se o valor de `aws:SourceArn` não contiver o ID da conta, como um ARN de bucket do Amazon S3, você deverá usar ambas as chaves de contexto de condição global para limitar as permissões.

O valor de `aws:SourceArn` deve ser a configuração para a qual o MediaStore publica os registros do CloudWatch em sua Região e conta.

O exemplo a seguir mostra como é possível usar as chaves de contexto de condição globais `aws:SourceArn` e `aws:SourceAccount` no MediaStore para evitar o problema confused deputy.

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "service.amazonaws.com"
    },
    "Action": "service:ActionName",
    "Resource": [
      "arn:aws:service::ResourceName/*"
    ],
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:service:*:123456789012:*"
      },
      "StringEquals": {
        "aws:SourceAccount": "123456789012"
      }
    }
  }
}
```

Monitoramento e marcação no AWS Elemental MediaStore

O monitoramento é uma parte importante para manter a confiabilidade, a disponibilidade e o desempenho do AWS Elemental MediaStore e das outras soluções da AWS. A AWS fornece as ferramentas de monitoramento a seguir para observar o MediaStore, informar quando algo está errado e realizar ações automaticamente quando apropriado:

- O AWS CloudTrail captura chamadas de API e eventos relacionados feitos por ou em nome de sua conta da AWS e entrega os arquivos de log a um bucket do Amazon S3 que você especificar. Você pode identificar quais usuários e contas chamaram a AWS, o endereço IP de origem do qual as chamadas foram feitas e quando elas ocorreram. Para obter mais informações, consulte o [Guia do usuário do AWS CloudTrail](#).
- O Amazon CloudWatch monitora os recursos da AWS e as aplicações que você executa na 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 fazer o CloudWatch acompanhar o uso da CPU ou outras métricas das instâncias do Amazon EC2 e iniciar automaticamente novas instâncias quando necessário. Para obter mais informações, consulte o [Guia do usuário do Amazon CloudWatch](#).
- O Amazon CloudWatch Events oferece uma transmissão de eventos do sistema que descrevem as mudanças nos recursos da AWS. Em geral, os serviços AWS entregam notificações de eventos ao CloudWatch Events em segundos, mas às vezes podem levar um minuto ou mais. O CloudWatch Events habilita a computação orientada a eventos automatizada, já que é possível escrever regras que monitoram determinados eventos e acionam ações automatizadas em outros serviços da AWS quando esses eventos ocorrem. Para obter mais informações, consulte o [Manual do usuário do Amazon CloudWatch Events](#).
- O Amazon CloudWatch Logs permite monitorar, armazenar e acessar os arquivos de log de instâncias do Amazon EC2, do CloudTrail e de outras fontes. O CloudWatch Logs pode monitorar informações nos arquivos de log e notificar você quando determinados limites forem atingidos. Você também pode arquivar seus dados de log em armazenamento resiliente. Para obter mais informações, consulte o [Guia do usuário do Amazon CloudWatch Logs](#).

Também é possível atribuir metadados aos contêineres do MediaStore na forma de tags. Cada tag é um rótulo que consiste em uma chave e um valor definidos por você. As tags podem facilitar o gerenciamento, a pesquisa e a filtragem de recursos. É possível usar tags para organizar os recursos

da AWS no Console de Gerenciamento da AWS, criar relatórios de uso e faturamento em todos os recursos da AWS e filtrar recursos durante atividades de automação de infraestrutura.

Tópicos

- [Registrar chamadas de API do AWS Elemental MediaStore com o AWS CloudTrail](#)
- [Monitorar o AWS Elemental MediaStore com o Amazon CloudWatch](#)
- [Marcação de recursos do AWS Elemental MediaStore](#)

Registrar chamadas de API do AWS Elemental MediaStore com o AWS CloudTrail

O AWS Elemental MediaStore é integrado ao AWS CloudTrail, um serviço que fornece um registro das ações executadas por um usuário, por um perfil ou por um serviço do AWS no MediaStore. O CloudTrail captura um subconjunto de chamadas de API para o MediaStore como eventos, incluindo as chamadas do console do MediaStore e de chamadas de código para a API do MediaStore. Se você criar uma trilha, poderá habilitar a entrega contínua de eventos do CloudTrail para um bucket do Amazon S3, incluindo eventos para o MediaStore. Se você não configurar uma trilha, ainda poderá visualizar os eventos mais recentes no console do CloudTrail em Event history (Histórico de eventos). Ao usar as informações coletadas pelo CloudTrail, é possível determinar a solicitação feita ao MediaStore, o endereço IP do qual a solicitação foi feita, quem a fez e quando ela foi feita e muito mais.

Para saber mais sobre o CloudTrail, incluindo como configurá-lo e ativá-lo, consulte o [Guia do usuário do AWS CloudTrail](#).

Tópicos

- [Informações sobre o AWS Elemental MediaStore no CloudTrail](#)
- [Exemplo: entradas do arquivo de log do AWS Elemental MediaStore](#)

Informações sobre o AWS Elemental MediaStore no CloudTrail

O CloudTrail é habilitado em sua conta da AWS quando ela é criada. Quando uma atividade de evento com suporte ocorre no AWS Elemental MediaStore, ela é registrada em um evento do CloudTrail juntamente com outros eventos de serviços da AWS no Histórico de eventos. Você pode visualizar, pesquisar e baixar eventos recentes em sua conta da AWS. Para obter mais informações, consulte [Como visualizar eventos com o histórico de eventos do CloudTrail](#).

Para obter um registro contínuo de eventos da sua conta da AWS, incluindo aqueles do MediaStore, crie uma trilha. Uma trilha permite que o CloudTrail entregue arquivos de log a um bucket do Amazon S3. Por padrão, quando você cria uma trilha no console, ela é aplicada a todas as regiões da AWS. A trilha registra em log eventos de todas as regiões na partição da AWS e entrega os arquivos de log para o bucket do Amazon S3 especificado por você. Além disso, é possível configurar outros serviços da AWS para analisar mais ainda mais e agir com base nos dados de eventos coletados nos logs do CloudTrail. Para obter mais informações, consulte os tópicos a seguir:

- [Visão geral da criação de uma trilha](#)
- [Serviços e integrações compatíveis com o CloudTrail](#)
- [Configurar notificações do Amazon SNS para o CloudTrail](#)
- [Receber arquivos de log do CloudTrail de várias regiões](#) e [receber arquivos de log do CloudTrail de várias contas](#)

O AWS Elemental MediaStore oferece suporte ao registro das ações a seguir como eventos nos arquivos de log do CloudTrail:

- [CreateContainer](#)
- [DeleteContainer](#)
- [DeleteContainerPolicy](#)
- [DeleteCorsPolicy](#)
- [DescribeContainer](#)
- [GetContainerPolicy](#)
- [GetCorsPolicy](#)
- [ListContainers](#)
- [PutContainerPolicy](#)
- [PutCorsPolicy](#)

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar:

- Se a solicitação foi feita com credenciais de usuário raiz ou credenciais de usuário
- Se a solicitação foi feita com credenciais de segurança temporárias de uma função ou de um usuário federado

- Se a solicitação foi feita por outro serviço da AWS

Para obter mais informações, consulte o [Elemento userIdentity do CloudTrail](#).

Exemplo: entradas do arquivo de log do AWS Elemental MediaStore

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log a um bucket do Amazon S3 especificado. Os arquivos de log do CloudTrail contêm uma ou mais entradas de log. Um evento representa uma única solicitação de qualquer origem e inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros de solicitação e assim por diante. Os arquivos de log do CloudTrail não são um rastreamento de pilha ordenada das chamadas de API pública. Dessa forma, eles não são exibidos em uma ordem específica.

O seguinte exemplo a seguir mostra uma entrada de log do CloudTrail que demonstra a operação `CreateContainer`:

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "ABCDEFGHIJKL123456789",
    "arn": "arn:aws:iam::111122223333:user/testUser",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "testUser",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2018-07-09T12:55:42Z"
      }
    }
  },
  "invokedBy": "signin.amazonaws.com"
},
{
  "eventTime": "2018-07-09T12:56:54Z",
  "eventSource": "mediastore.amazonaws.com",
  "eventName": "CreateContainer",
  "awsRegion": "ap-northeast-1",
  "sourceIPAddress": "54.239.119.16",
  "userAgent": "signin.amazonaws.com",
  "requestParameters": {
    "containerName": "TestContainer"
  }
}
```



```
    },
    "responseElements": {
      "container": {
        "status": "CREATING",
        "creationTime": "Jul 9, 2018 12:56:54 PM",
        "name": " TestContainer ",
        "aRN": "arn:aws:mediastore:ap-northeast-1:111122223333:container/
TestContainer"
      }
    },
    "requestID":
    "MNCTGH4HRQJ27GRMBVDPIVHEP4L02BN6MUVHBCPSHOAWNSOKSXC024B2UE0BBND5D0NRXTMFK3TOJ4G7AHWMESI",
    "eventID": "7085b140-fb2c-409b-a329-f567912d704c",
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
  }
}
```

Monitorar o AWS Elemental MediaStore com o Amazon CloudWatch

Você pode monitorar o AWS Elemental MediaStore usando o Amazon CloudWatch, que coleta dados brutos e os processa em métricas legíveis. O CloudWatch mantém estatísticas 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 do usuário do Amazon CloudWatch](#).

A AWS fornece as seguintes ferramentas de monitoramento para supervisionar o MediaStore, informar quando algo está errado e realizar ações automáticas quando apropriado:

- O Amazon CloudWatch Logs permite monitorar, armazenar e acessar os arquivos de log de serviços da AWS como o AWS Elemental MediaStore. Você pode usar o CloudWatch Logs para monitorar aplicativos e sistemas com dados de log. Por exemplo, o CloudWatch Logs pode monitorar o número de erros que ocorrerem em seus logs de aplicação e enviar uma notificação sempre que a taxa de erros exceder um limite especificado. Como o CloudWatch Logs usa seus dados de log para monitoramento, nenhuma alteração de código é necessária. Por exemplo, você pode monitorar os logs de aplicativo para termos literais específicos (como "ValidationException") ou contar o número de solicitações `PutObject` feitas durante um determinado período. Quando o termo que você estiver procurando for encontrado, o CloudWatch Logs relatará os dados para uma

métrica do CloudWatch que você especificar. Os dados de log são criptografados em trânsito e em repouso.

- O Amazon CloudWatch Events fornece eventos do sistema que descrevem as alterações nos recursos da AWS, como objetos do MediaStore. Em geral, os serviços AWS entregam notificações de eventos ao CloudWatch Events em segundos, mas às vezes podem levar um minuto ou mais. Você pode configurar regras para corresponder a eventos (como uma solicitação `DeleteObject`) e roteá-los para um ou mais streams ou funções de destino. O CloudWatch Events se torna ciente das alterações operacionais no momento em que ocorrem. Além disso, o CloudWatch Events responde a essas alterações operacionais e executa a ação corretiva conforme necessário, enviando mensagens para responder ao ambiente, ativando funções, fazendo alterações e capturando informações de estado.

CloudWatch Logs

O registro de acesso em log fornece detalhes sobre as solicitações que são feitas aos objetos de um contêiner. Os logs de acesso são úteis para muitas aplicações, como auditorias de segurança e acesso. Eles também podem ajudar a conhecer sua base de clientes e entender sua fatura do MediaStore. Os Logs do CloudWatch são classificados da seguinte forma:

- Uma transmissão de log é uma sequência de eventos de log que compartilham a mesma fonte.
- Um grupo logs é um grupo de fluxos de log que compartilham as mesmas configurações de retenção, monitoramento e controle de acesso. Ao habilitar o registro de acesso em um contêiner, o MediaStore cria um grupo de logs com um nome como `/aws/mediastore/MyContainerName`. Você pode definir grupos de logs e especificar quais fluxos colocar em cada grupo. Não há cota para o número de streams de log que podem pertencer a um grupo de logs.

Por padrão, os logs são mantidos indefinidamente e nunca expiram. Você pode ajustar a política de retenção para cada grupo de logs, mantendo a retenção indefinida ou escolhendo um período de retenção de um dia a 10 anos.

Configuração de permissões para o Amazon CloudWatch

Use o AWS Identity and Access Management (IAM) para criar um perfil que dê ao AWS Elemental MediaStore acesso ao Amazon CloudWatch. Você deve executar estas etapas para publicar o CloudWatch Logs na sua conta. O CloudWatch publica automaticamente métricas para sua conta.

Para permitir que o MediaStore acesse o CloudWatch

1. Abra o console do IAM em <https://console.aws.amazon.com/iam/>.
2. No painel de navegação do console do IAM, escolha Políticas (Políticas) e Create policy (Criar política).
3. Selecione a guia JSON e cole a política a seguir:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:DescribeLogGroups",
        "logs:CreateLogGroup"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogStream",
        "logs:DescribeLogStreams",
        "logs:PutLogEvents"
      ],
      "Resource": "arn:aws:logs:*:*:log-group:/aws/mediastore/*"
    }
  ]
}
```

Esta política permite que o MediaStore crie grupos de logs e fluxos de logs para todos os contêineres em qualquer região na sua conta da AWS.

4. Escolha Review policy (Revisar política).
5. Na página Review policy (Revisar política), em Name (Nome), digite **MediaStoreAccessLogsPolicy** e escolha Create policy (Criar política).
6. No painel de navegação do console do IAM, escolha Roles (Funções) e, em seguida, Create role (Criar função).
7. Escolha o tipo de função Another AWS account (Outra conta da AWS).
8. Em Account ID (ID da conta), digite o ID da conta da AWS.

9. Escolha Next: Permissions (Próximo: permissões).
10. Na caixa de pesquisa, insira **MediaStoreAccessLogsPolicy**.
11. Marque a caixa de seleção ao lado de sua nova política e escolha Next: Tags (Próximo: Tags).
12. Escolha Next: Review (Próximo: rever) para visualizar o novo usuário.
13. Em Role name (Nome da função), digite **MediaStoreAccessLogs** e escolha Create role (Criar função).
14. Na mensagem de confirmação, escolha o nome da função que você acabou de criar (**MediaStoreAccessLogs**).
15. Na página Summary (Resumo) da função, escolha a guia Trust relationship (Relação de confiança).
16. Escolha Edit trust relationship (Editar relação de confiança).
17. No documento da política, altere a entidade principal para o serviço MediaStore. A aparência deve ser semelhante a esta:

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

Toda a política deve ser lida da seguinte maneira:

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

18. Escolha Update Trust Policy.

Habilitar registro em log de acessos para um contêiner

Por padrão, o AWS Elemental MediaStore não coleta logs de acesso. Ao habilitar o registro de acesso em log em um contêiner, o MediaStore fornece logs de acesso para objetos armazenados no contêiner ao Amazon CloudWatch. Os logs de acesso fornecem registros detalhados para solicitações feitas para qualquer objeto armazenado no contêiner. Essa informação pode incluir o tipo de solicitação, os recursos que foram especificados na solicitação e a hora e data em que a solicitação foi processada.

Important

Não há custo adicional para a habilitação do registro de acesso em um contêiner do MediaStore. No entanto, todos os arquivos de log que o serviço fornece acumulam as cobranças normais de armazenamento. (Você pode excluir os arquivos de log a qualquer momento.) A AWS não avalia cobranças de transferência de dados para fornecimento de arquivos de log, mas cobra a taxa de transferência de dados normal para acessar os arquivos de log.

Para habilitar o registro de acesso em logs (AWS CLI)

- Na AWS CLI, use o comando `start-access-logging`:

```
aws mediastore start-access-logging --container-name LiveEvents --region us-west-2
```

Este comando não possui valor de retorno.

Desabilitar o registro em log de acessos para um contêiner

Ao desabilitar o registro de acesso em logs em um contêiner, o AWS Elemental MediaStore interrompe o envio de logs de acesso ao Amazon CloudWatch. Esses logs de acesso não são salvos e não são recuperáveis.

Para desabilitar o registro de acesso em log (AWS CLI)

- Na AWS CLI, use o comando `stop-access-logging`:

```
aws mediastore stop-access-logging --container-name LiveEvents --region us-west-2
```

Este comando não possui valor de retorno.

Solução de problemas de registro de acesso no AWS Elemental MediaStore

Quando os logs de acesso do AWS Elemental MediaStore não aparecerem no Amazon CloudWatch, consulte a tabela a seguir para saber as possíveis causas e resoluções.

Note

Certifique-se de habilitar os logs do AWS CloudTrail para ajudar com o processo de solução de problemas.

Sintomas	O problema pode ser...	Experimente Isso...
Você não vê nenhum evento do CloudTrail, mesmo com os logs do CloudTrail ativados.	A perfil do IAM não existe ou o nome, as permissões ou a política de confiança estão incorretos.	Crie uma função com o nome, permissões e política de confiança corretos. Consulte the section called “Configuração de permissões para o CloudWatch” .
Você enviou uma solicitação de API <code>DescribeContainer</code> , mas a resposta mostra que o parâmetro <code>AccessLoggingEnabled</code> tem um valor de <code>False</code> . Além disso, você não consegue visualizar nenhum evento do CloudTrail para o perfil do <code>MediaStoreAccessLogs</code> fazendo uma chamada bem-sucedida de <code>DescribeLogGroup</code> , <code>CreateLogGroup</code> , <code>DescribeLogStream</code> ou <code>CreateLogStream</code> .	A perfil do IAM não existe ou o nome, as permissões ou a política de confiança estão incorretos. O registro de acesso em logs não está ativado no contêiner.	Crie uma função com o nome, permissões e política de confiança corretos. Consulte the section called “Configuração de permissões para o CloudWatch” . Habilitar logs de acesso para o contêiner. Consulte the section called “Habilitar registro em log de acessos” .

Sintomas	O problema pode ser...	Experimente Isso...
<p>No console do CloudTrail, você verá um evento com um erro de acesso negado relacionado ao perfil do MediaStoreAccessLogs . O evento do CloudTrail pode incluir linhas como as seguintes:</p> <pre>"eventSource": "logs.amazonaws.com", "errorCode": "AccessDenied", "errorMessage": "User: arn:aws:sts::11112223333:assumed-role/MediaStoreAccessLogs/MediaStoreAccessLogsSession is not authorized to perform: logs:DescribeLogGroups on resource: arn:aws:logs:us-west-2:11112223333:log-group::log-stream:",</pre>	<p>A perfil do IAM não tem as permissões corretas para o AWS Elemental MediaStore.</p>	<p>Atualize a perfil do IAM para ter as permissões e política de confiança corretas. Consulte the section called “Configuração de permissões para o CloudWatch”.</p>

Sintomas	O problema pode ser...	Experimente Isso...
Você não vê nenhum log para um contêiner inteiro ou contêineres.	Sua conta pode ter excedido a cota do CloudWatch para grupos de logs por conta, por região. Veja as cotas para grupos de log no Guia do usuário do Amazon CloudWatch Logs .	No console do CloudWatch, determine se sua conta atingiu a cota do CloudWatch para grupos de logs. Se necessário, solicite um aumento da cota .
Alguns logs são vistos no CloudWatch, mas nem todos os que você espera.	Sua conta pode ter excedido a cota do CloudWatch de transações por segundo por conta, por região. Veja as cotas para PutLogEvents no Guia do usuário do Amazon CloudWatch Logs .	Solicite um aumento da cota para transações do CloudWatch por segundo por conta, por Região.

Formato de log de acessos

Os arquivos de log de acesso consistem em uma sequência de registros de log em formato JSON, em que cada registro de log representa uma solicitação. A ordem dos campos dentro do log pode variar. Veja a seguir um exemplo de log que consiste em dois registros de log:

```
{
  "Path": "/FootballMatch/West",
  "Requester": "arn:aws:iam::111122223333:user/maria-garcia",
  "AWSAccountId": "111122223333",
```



```

"RequestID":
"aaaAAA111bbbBBB222cccCCC333dddDDD444eeeEEE555ffffFFF666gggGGG777hhhHHH888iiiIII999jjjJJJ",
"ContainerName": "LiveEvents",
"TotalTime": 147,
"BytesReceived": 1572864,
"BytesSent": 184,
"ReceivedTime": "2018-12-13T12:22:06.245Z",
"Operation": "PutObject",
"ErrorCode": null,
"Source": "192.0.2.3",
"HTTPStatus": 200,
"TurnAroundTime": 7,
"ExpiresAt": "2018-12-13T12:22:36Z"
}
{
"Path": "/FootballMatch/West",
"Requester": "arn:aws:iam::111122223333:user/maria-garcia",
"AWSAccountId": "111122223333",
"RequestID":
"dddDDD444eeeEEE555ffffFFF666gggGGG777hhhHHH888iiiIII999jjjJJJ000cccCCC333bbbBBB222aaaAAA",
"ContainerName": "LiveEvents",
"TotalTime": 3,
"BytesReceived": 641354,
"BytesSent": 163,
"ReceivedTime": "2018-12-13T12:22:51.779Z",
"Operation": "PutObject",
"ErrorCode": "ValidationException",
"Source": "198.51.100.15",
"HTTPStatus": 400,
"TurnAroundTime": 1,
"ExpiresAt": null
}

```

A lista a seguir descreve os campos dos registros de log:

AWSAccountId

O ID de conta da AWS da conta que foi usada para fazer a solicitação.

BytesReceived

O número de bytes no corpo da solicitação recebidos pelo servidor do MediaStore.

BytesSent

O número de bytes no corpo da resposta enviados pelo servidor do MediaStore. Esse valor geralmente é o mesmo que o valor do cabeçalho Content-Length incluído com respostas do servidor.

ContainerName

O nome do contêiner que recebeu a solicitação.

ErrorCode

O código de erro do MediaStore (como `InternalServerError`). Se nenhum erro tiver ocorrido, o caractere - será exibido. Um código de erro poderá ser exibido mesmo se o código de status for 200 (o que indica uma conexão fechada ou um erro depois de o servidor iniciar a transmissão da resposta).

ExpiresAt

A data e a hora de expiração do objeto. Esse valor é baseado na idade de expiração definida por [transient data rule](#) na política de ciclo de vida aplicada ao contêiner. O valor é horário e data ISO-8601 baseados no relógio do sistema do host que atendeu a solicitação. Se a política de ciclo de vida não tiver uma regra de dados transitória que se aplique ao objeto, ou se não houver uma política de ciclo de vida aplicada ao contêiner, o valor desse campo será `null`. Esse campo será aplicado apenas às seguintes operações: `PutObject`, `GetObject`, `DescribeObject` e `DeleteObject`.

HTTPStatus

O código numérico do status do HTTP da resposta.

Operação

A operação executada, como `PutObject` ou `ListItems`.

Path

O caminho dentro do contêiner no qual o objeto está armazenado. Se a operação não leva um parâmetro de caminho, o caractere - é exibido.

ReceivedTime

A hora do dia em que a solicitação foi recebida. O valor é horário e data ISO-8601 baseados no relógio do sistema do host que atendeu a solicitação.

Solicitante

O usuário Nome de recurso da Amazon (ARN) da conta que foi usada para fazer a solicitação. Para solicitações não autenticadas, esse valor será `anonymous`. Se a solicitação falhar antes de a autenticação ser concluída, esse campo poderá estar ausente do log. Para tais solicitações, o `ErrorCode` pode identificar o problema de autorização.

RequestID

Uma string gerada pelo AWS Elemental MediaStore para identificar exclusivamente cada solicitação.

Origem

O endereço da Internet aparente do solicitante ou do gerente do serviço da AWS que está fazendo a chamada. Se os firewalls e proxies intermediários obscurecerem o endereço da máquina que fez a solicitação, o valor será definido como nulo.

TotalTime

O número de milissegundos (ms) em que a solicitação esteve em andamento da perspectiva do servidor. Esse valor é medido a partir do momento em que o pedido é recebido pelo serviço e termina no momento em que o último byte da resposta é enviado. Esse valor é medido do ponto de vista do servidor porque medições feitas da perspectiva do cliente são afetadas pela latência de rede.

TurnAroundTime

O número de milissegundos que o MediaStore gastou para processar a solicitação. Esse valor é medido do momento do recebimento do último byte da solicitação até o momento em que o primeiro byte da resposta foi enviado.

A ordem dos campos no log pode variar.

As alterações do status dos registros em log entram em vigor ao longo do tempo

As alterações no status do registro em log de um contêiner levam tempo para realmente afetar a entrega de arquivos de log. Por exemplo, se você habilitar o registro em log para um contêiner A, algumas solicitações feitas na hora seguinte podem ser registradas, enquanto outras não. Se você desabilitar o registro em log para o contêiner B, alguns logs poderão continuar a ser entregues na hora seguinte, enquanto outros não. Em todos os casos, as novas configurações acabam sendo aplicadas sem que você necessite tomar qualquer outra ação.

Entrega de logs do servidor de melhor esforço

Os registros de log de acessos são entregues com base no melhor esforço. A maioria das solicitações para um contêiner configurado corretamente para registro em log tem como resultado um registro do log entregue. A maioria dos registros de log é entregue dentro de algumas horas após o tempo em que forem registrados, mas eles podem ser entregues com mais frequência.

A integralidade e a pontualidade do registro de acesso em log não são garantidas. O registro de log de uma solicitação específica pode ser entregue muito depois de a solicitação ter sido realmente processada ou pode nem ser entregue. A finalidade dos logs de acesso é proporcionar uma ideia da natureza do tráfego no contêiner. É raro perder registros de log, mas o registro de acesso em logs não tem como objetivo ser uma contabilidade completa de todas as solicitações.

Levando em conta a natureza de melhor esforço do atributo de registro de acesso em log, os relatórios de uso disponíveis no portal da AWS (relatórios do Gerenciamento de custos e faturamento no [AWS Management Console](#)) podem incluir uma ou mais solicitações de acesso que não aparecem em um log de acesso entregue.

Considerações de programação para o formato de logs de acesso

Periodicamente, você pode estender o formato de log de acesso adicionando novos campos. O código que analisa os logs de acesso devem ser escritos para tratar de campos adicionais que não compreende.

CloudWatch Events

O Amazon CloudWatch Events permite que você automatize seus serviços da AWS e responda automaticamente a eventos do sistema, como problemas de disponibilidade de aplicação ou alterações de recursos. Você pode escrever regras simples para indicar quais eventos são do seu interesse, e as ações automatizadas a serem tomadas quando um evento corresponder à regra.

Important

Em geral, os serviços AWS entregam notificações de eventos ao CloudWatch Events em segundos, mas às vezes podem levar um minuto ou mais.

Quando um arquivo é carregado em um contêiner ou removido dele, dois eventos são acionados em sucessão no serviço do CloudWatch:

1. [the section called “Evento de alteração de estado do objeto”](#)
2. [the section called “Evento de alteração de estado do contêiner”](#)

Para obter informações sobre como se inscrever nesses eventos, consulte [Amazon CloudWatch](#).

As ações que podem ser automaticamente acionadas incluem as seguintes:

- Como invocar uma função do AWS Lambda
- Invocação do Run Command do Amazon EC2
- Retransmissão do evento para o Amazon Kinesis Data Streams
- Ativação da máquina de estado do AWS Step Functions
- Notificação de um tópico do Amazon SNS ou de uma fila do AWS SMS

Alguns exemplos de uso do CloudWatch Events com o AWS Elemental MediaStore incluem os seguintes:

- Ativar uma função do Lambda sempre que um contêiner é criado
- Notificar um tópico do Amazon SNS quando um objeto é excluído

Para obter mais informações, consulte o [Manual do usuário do Amazon CloudWatch Events](#).

Tópicos

- [Evento de mudança de estado de objeto do AWS Elemental MediaStore](#)
- [Evento de mudança de estado do contêiner do AWS Elemental MediaStore](#)

Evento de mudança de estado de objeto do AWS Elemental MediaStore

Esse evento é publicado quando o estado de um objeto for alterado, quando o objeto for carregado ou excluído.

Note

Objetos que expiram devido a uma regra de dados transitória não emitem um evento do CloudWatch ao expirarem.

Para obter informações sobre como se inscrever nesse evento, consulte [Amazon CloudWatch](#).

Objeto atualizado

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Object State Change",
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:mediastore:us-east-1:111122223333:MondayMornings/Episode1/Introduction.avi"
  ],
  "detail": {
    "ContainerName": "Movies",
    "Operation": "UPDATE",
    "Path": "TVShow/Episode1/Pilot.avi",
    "ObjectSize": 123456,
    "URL": "https://a832p1qeaznlp9.files.mediastore-us-west-2.com/Movies/MondayMornings/Episode1/Introduction.avi"
  }
}
```

Objeto removido

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Object State Change",
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:mediastore:us-east-1:111122223333:Movies/MondayMornings/Episode1/Introduction.avi"
  ],
  "detail": {
    "ContainerName": "Movies",
    "Operation": "REMOVE",
  }
}
```

```
"Path": "Movies/MondayMornings/Episode1/Introduction.avi",
"URL": "https://a832p1qeaznlp9.files.mediastore-us-west-2.com/Movies/
MondayMornings/Episode1/Introduction.avi"
}
}
```

Evento de mudança de estado do contêiner do AWS Elemental MediaStore

Esse evento é publicado quando o estado de um contêiner for alterado, quando um contêiner for adicionado ou excluído. Para obter informações sobre como se inscrever nesse evento, consulte [Amazon CloudWatch](#).

Contêiner criado

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Container State Change",
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:mediastore:us-east-1:111122223333:container/Movies"
  ],
  "detail": {
    "ContainerName": "Movies",
    "Operation": "CREATE"
    "Endpoint": "https://a832p1qeaznlp9.mediastore-us-west-2.amazonaws.com"
  }
}
```

Contêiner removido

```
{
  "version": "1",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "MediaStore Container State Change",
  "source": "aws.mediastore",
  "account": "111122223333",
  "time": "2017-02-22T18:43:48Z",
  "region": "us-east-1",
```

```
"resources": [  
  "arn:aws:mediastore:us-east-1:111122223333:container/Movies"  
],  
"detail": {  
  "ContainerName": "Movies",  
  "Operation": "REMOVE"  
}  
}
```

Monitorar o AWS Elemental MediaStore com métricas do Amazon CloudWatch

Você pode monitorar o AWS Elemental MediaStore usando o Amazon CloudWatch, que coleta dados brutos e os processa em métricas legíveis. O CloudWatch mantém 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 do usuário do Amazon CloudWatch](#).

Para o AWS Elemental MediaStore, talvez você queira observar BytesDownloaded e enviar um e-mail para si mesmo quando essa métrica atingir um determinado limite.

Para exibir métricas usando o console do CloudWatch

As métricas são agrupadas primeiro pelo namespace do serviço e, em seguida, por várias combinações de dimensão dentro de cada namespace.

1. Faça login no AWS Management Console e abra o console do CloudWatch em <https://console.aws.amazon.com/cloudwatch/>.
2. No painel de navegação, escolha Metrics (Métricas).
3. Em All metrics (Todas as métricas), escolha o namespace AWS/MediaStore.
4. Escolha a dimensão métrica para visualizar as métricas. Por exemplo, escolha Request metrics by container para visualizar métricas para os diferentes tipos de solicitações que foram enviadas para o contêiner.

Para visualizar métricas usando o AWS CLI

- Em um prompt de comando, use o seguinte comando:


```
aws cloudwatch list-metrics --namespace "AWS/MediaStore"
```

Métricas do AWS Elemental MediaStore

A tabela a seguir lista as métricas que o AWS Elemental MediaStore envia para o CloudWatch.

Note

Para visualizar as métricas, é necessário [adicionar uma política de métrica](#) ao contêiner para permitir que o MediaStore envie métricas ao Amazon CloudWatch.

Métrica	Descrição
RequestCount	<p>O número total de solicitações HTTP feitas para um contêiner do MediaStore separadas por tipo de operação (Put, Get, Delete, Describe e List).</p> <p>Unidade: contagem</p> <p>Dimensões válidas:</p> <ul style="list-style-type: none"> • Nome do contêiner • Nome do grupo de objetos • Tipos de solicitação <p>Estatística válida: soma</p>
4xxErrorCount	<p>O número de solicitações HTTP feitas para o MediaStore que resultou em um erro 4xx.</p> <p>Unidade: contagem</p> <p>Dimensões válidas:</p> <ul style="list-style-type: none"> • Nome do contêiner • Nome do grupo de objetos • Tipos de solicitação

Métrica	Descrição
	Estatística válida: soma
5xxErrorCount	<p>O número de solicitações HTTP feitas para o MediaStore que resultou em um erro 5xx.</p> <p>Unidade: contagem</p> <p>Dimensões válidas:</p> <ul style="list-style-type: none">• Nome do contêiner• Nome do grupo de objetos• Tipos de solicitação <p>Estatística válida: soma</p>
BytesUploaded	<p>O número de bytes carregados para solicitações feitas a um contêiner do MediaStore, em que a solicitação inclui um corpo.</p> <p>Unidade: bytes</p> <p>Dimensões válidas:</p> <ul style="list-style-type: none">• Nome do contêiner• Nome do grupo de objetos <p>Estatísticas válidas: média (bytes por solicitação), soma (bytes por período), contagem de amostras, mín. (o mesmo que P0,0), máx. (o mesmo que P100), qualquer percentil entre P0,0 e P99,9</p>

Métrica	Descrição
BytesDownloaded	<p>O número de bytes obtidos por download para solicitações feitas a um contêiner do MediaStore, em que a resposta inclui um corpo.</p> <p>Unidade: bytes</p> <p>Dimensões válidas:</p> <ul style="list-style-type: none">• Nome do contêiner• Nome do grupo de objetos <p>Estatísticas válidas: média (bytes por solicitação), soma (bytes por período), contagem de amostras, mín. (o mesmo que P0,0), máx. (o mesmo que P100), qualquer percentil entre P0,0 e P99,9</p>
TotalTime	<p>O número de milissegundos (ms) em que a solicitação esteve em trânsito da perspectiva do servidor. Esse valor é medido a partir do momento que o MediaStore recebe sua solicitação até o momento que ele envia o último byte da resposta. Esse valor é medido do ponto de vista do servidor porque medições feitas da perspectiva do cliente são afetadas pela latência de rede.</p> <p>Unidade: milissegundos</p> <p>Dimensões válidas:</p> <ul style="list-style-type: none">• Nome do contêiner• Nome do grupo de objetos• Tipos de solicitação <p>Estatísticas válidas: média, mín. (igual a P0,0), máx. (o mesmo que P100), qualquer percentil entre P0,0 e P100</p>

Métrica	Descrição
TurnaroundTime	<p>O número de milissegundos que o MediaStore gastou para processar a solicitação. Este valor é medido a partir do momento que o MediaStore recebe o último byte da sua solicitação, até o momento que envia o primeiro byte da resposta.</p> <p>Unidade: milissegundos</p> <p>Dimensões válidas:</p> <ul style="list-style-type: none"> • Nome do contêiner • Nome do grupo de objetos • Tipos de solicitação <p>Estatísticas válidas: média, mín. (igual a P0,0), máx. (o mesmo que P100), qualquer percentil entre P0,0 e P100</p>
ThrottleCount	<p>O número de solicitações HTTP feitas ao MediaStore que sofreram controle de utilização.</p> <p>Unidade: contagem</p> <p>Dimensões válidas:</p> <ul style="list-style-type: none"> • Nome do contêiner • Nome do grupo de objetos • Tipos de solicitação <p>Estatística válida: soma</p>

Marcação de recursos do AWS Elemental MediaStore

Uma tag é um rótulo de atributo personalizado que você ou a AWS atribui a um recurso da AWS. Cada tag tem duas partes:

- Uma chave de tag (por exemplo CostCenter, Environment ou Project). Chaves de tag fazem distinção entre maiúsculas e minúsculas.
- Um campo opcional conhecido como um valor de tag (por exemplo, 111122223333 ou Production). Omitir o valor da tag é o mesmo que usar uma string vazia. Como chaves de tag, os valores das tags diferenciam maiúsculas de minúsculas.

As tags ajudam você a fazer o seguinte:

- Identificar e organizar seus recursos da AWS. Muitos serviços da AWS oferecem suporte à marcação para que você possa atribuir a mesma tag a recursos de diferentes serviços para indicar que os recursos estão relacionados. Por exemplo, é possível atribuir a mesma tag a um *contêiner* do AWS Elemental MediaStore que você atribui a uma entrada do AWS Elemental MediaLive.
- Monitorar seus custos da AWS. Você ativa essas tags no painel do AWS Billing and Cost Management. A AWS usa as tags para categorizar seus custos e fornecer um relatório mensal de alocação de custos mensais a você. Para obter mais informações, consulte [Usar tags de alocação de custos](#) no [Guia do usuário do AWS Billing](#).

As seções a seguir fornecem mais informações sobre tags para o AWS Elemental MediaStore.

Recursos compatíveis no AWS Elemental MediaStore

Os recursos a seguir no AWS Elemental MediaStore são compatíveis com a marcação:

- *contêiner*

Para obter informações sobre como adicionar e gerenciar tags, consulte [Gerenciar tags](#).

O AWS Elemental MediaStore não é compatível com o atributo de controle de acesso baseado em tags do AWS Identity and Access Management (IAM).

Convenções de uso e nomenclatura de tags

As seguintes convenções básicas de uso e nomenclatura se aplicam ao uso de tags com recursos do AWS Elemental MediaStore:

- Cada recurso pode ter um máximo de 50 tags.

- Em todos os recursos, cada chave de tag deve ser exclusiva e pode ter apenas um valor.
- O comprimento máximo da chave da tag é de 128 caracteres Unicode em UTF-8.
- O comprimento máximo do valor da tag é de 256 caracteres Unicode em UTF-8.
- Os caracteres permitidos são letras, números, espaços representáveis em UTF-8, além dos seguintes caracteres: . : + = @ _ / - (hífen). Os recursos do Amazon EC2 permitem qualquer caractere.
- As chaves e os valores de tags diferenciam maiúsculas de minúsculas. Como melhor prática, adote uma estratégia para letras maiúsculas em tags e implemente-a de forma consistente em todos os tipos de recursos. Por exemplo, decida se deseja usar `Costcenter`, `costcenter` ou `CostCenter` e use a mesma convenção para todas as tags. Evite usar tags semelhantes com tratamento do tamanho de letra inconsistente.
- O prefixo `aws :` é proibido em tags, pois ele é reservado para uso pela AWS. Você não pode editar nem excluir chaves nem valores de tag com esse prefixo. As tags com esse prefixo não contam para as tags por cota de recurso.

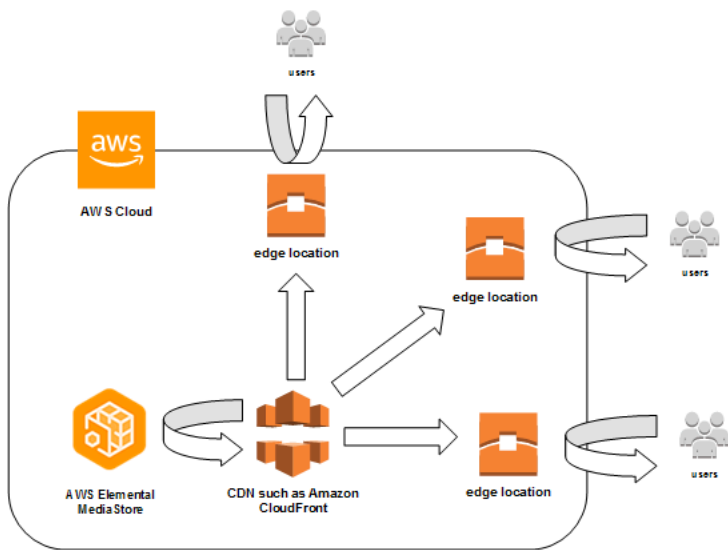
Gerenciar tags

As tags são compostas de propriedades `Value` e `Key` em um recurso. É possível usar a AWS CLI ou a API do MediaStore para adicionar, editar ou excluir os valores dessas propriedades. Para obter informações sobre como trabalhar com tags, consulte as seções a seguir na Referência de API do AWS Elemental MediaStore:

- [CreateContainer](#)
- [ListTagsForResource](#)
- [Recursos](#)
- [TagResource](#)
- [UntagResource](#)

Trabalhar com redes de entrega de conteúdo (CDNs)

Você pode usar uma rede de entrega de conteúdo (CDN), como o [Amazon CloudFront](#), para fornecer o conteúdo que você armazena no AWS Elemental MediaStore. Uma CDN é um conjunto globalmente distribuído de servidores que armazena conteúdo em cache como vídeos. Quando um usuário solicita o conteúdo, a CDN encaminha a solicitação para o local da borda que fornece a latência mais baixa. Se o conteúdo já está armazenado em cache nesse local da borda, a CDN o entrega imediatamente. Se o conteúdo não está nesse local da borda, a CDN o recupera de sua origem, como o contêiner do MediaStore, e o distribui ao usuário.



Tópicos

- [Permissão para o Amazon CloudFront acessar seu contêiner do AWS Elemental MediaStore](#)
- [Interação do AWS Elemental MediaStore com caches HTTP](#)

Permissão para o Amazon CloudFront acessar seu contêiner do AWS Elemental MediaStore

Você pode usar o Amazon CloudFront para fornecer o conteúdo que você armazena em um contêiner no AWS Elemental MediaStore. É possível fazer isso das seguintes maneiras:

- [Uso do controle de acesso de origem \(OAC\)](#): (recomendado) Use essa opção se o Região da AWS for compatível com o atributo OAC do CloudFront.

- [Uso de segredos compartilhados](#): use essa opção se sua Região da AWS não for compatível com o atributo OAC do CloudFront.

Uso do controle de acesso de origem (OAC)

Você pode usar o atributo de controle de acesso de origem (OAC) do Amazon CloudFront para proteger as origens do AWS Elemental MediaStore com segurança aprimorada. Você pode habilitar o [AWS Signature Version 4 \(SigV4\)](#) nas solicitações do CloudFront para origens do MediaStore e definir quando e se o CloudFront deve assinar as solicitações. Você pode acessar o atributo OAC do CloudFront por meio do console, de APIs, do SDK ou da CLI, e não há taxas de uso adicionais.

Para obter mais informações sobre o uso do atributo OAC com o MediaStore, consulte [Restringir o acesso a uma origem do MediaStore](#) no [Guia do desenvolvedor do Amazon CloudFront](#).

Uso de segredos compartilhados

Se a Região da AWS não oferece suporte ao atributo OAC do Amazon CloudFront, você pode anexar uma política ao contêiner do AWS Elemental MediaStore que conceda acesso de leitura ou superior ao CloudFront.

Note

Recomendamos usar o atributo OAC se a Região da AWS for compatível com ele. Os procedimentos a seguir exigem que você configure o MediaStore e o CloudFront com segredos compartilhados para restringir o acesso aos contêineres do MediaStore. Para seguir as práticas recomendadas de segurança, essa configuração manual exige uma rotação periódica de segredos. Com o OAC nas origens do MediaStore, você pode instruir o CloudFront a assinar solicitações usando o SigV4 e encaminhá-las ao MediaStore para correspondência de assinaturas, eliminando a necessidade de usar e alternar segredos. Isso garante a verificação automática das solicitações antes da vinculação do conteúdo de mídia, tornando a entrega de conteúdo de mídia por meio do MediaStore e do CloudFront mais simples e segura.

Habilitando o acesso do CloudFront ao contêiner (console)

1. Abra o console do MediaStore em <https://console.aws.amazon.com/mediastore/>.
2. Na página Containers (Contêineres), escolha o nome do contêiner.

A página de detalhes do contêiner é exibida.

3. Na seção Política de contêiner, anexe uma política que concede acesso de leitura ou maior acesso ao Amazon CloudFront.

Example

O exemplo de política a seguir, que é similar ao exemplo de política para [Acesso de leitura pública por HTTPS](#), está de acordo com esses requisitos, pois permite comandos `GetObject` e `DescribeObject` de qualquer pessoa que envia solicitações para seu domínio por meio de HTTPS. Além disso, o exemplo de política a seguir protege melhor seu fluxo de trabalho porque permite que o CloudFront acesse objetos do MediaStore apenas quando a solicitação ocorre por meio de uma conexão HTTPS e contém o cabeçalho `Referer` correto.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CloudFrontRead",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [
        "mediastore:GetObject",
        "mediastore:DescribeObject"
      ],
      "Resource": "arn:aws:mediastore:<region>:<owner acct
number>:container/<container name>/*",
      "Condition": {
        "StringEquals": {
          "aws:Referer": "<secretValue>"
        },
        "Bool": {
          "aws:SecureTransport": "true"
        }
      }
    }
  ]
}
```

4. Na seção Container CORS policy (Política de CORS de contêiner), atribua uma política que permita o nível de acesso apropriado.

Note

Uma [política de CORS](#) é necessária somente se você deseja conceder acesso a um jogador baseado em navegador.

5. Anote os seguintes detalhes:

- O endpoint de dados que é atribuído a seu contêiner do . Você pode encontrar essas informações na seção Info (Informações) da página Containers (Contêineres). No CloudFront, o endpoint de dados é denominado como nome de domínio de origem.
- A estrutura de pastas no contêiner em que os objetos são armazenados. No CloudFront, isso é chamado de caminho de origem. Essa configuração é opcional. Para obter mais informações sobre caminhos de origem, consulte o [Guia do desenvolvedor do Amazon CloudFront](#).

6. No CloudFront, crie uma distribuição [configurada para fornecer conteúdo do AWS Elemental MediaStore](#). Você precisará das informações coletadas na etapa anterior.

Após anexar a política aos seus contêineres do MediaStore, configure o CloudFront para usar apenas conexões HTTPS para solicitações de origem e também adicionar um cabeçalho personalizado com o valor secreto correto.

Configurando o CloudFront para acessar seu contêiner por meio de uma conexão HTTPS com um valor secreto para o cabeçalho Referer (console)

1. Abra o console do CloudFront.
2. Na página Origins (Origens), escolha sua origem no MediaStore.
3. Escolha Editar.
4. Selecione Somente HTTPS para o protocolo.
5. Na seção Adicionar cabeçalho personalizado, escolha Adicionar cabeçalho.
6. Para Nome, escolha Referer. Para o valor, use a mesma string `<secretValue>` usada na política de contêiner.
7. Escolha Salvar e deixe as alterações serem implantadas.

Interação do AWS Elemental MediaStore com caches HTTP

O AWS Elemental MediaStore armazena objetos para que eles possam ser armazenados em cache de forma correta e eficiente por redes de entrega de conteúdo (CDNs) como o Amazon CloudFront. Quando um usuário final ou CDN recupera um objeto do MediaStore, o serviço retorna cabeçalhos HTTP que afetam o comportamento do armazenamento em cache do objeto. (Os padrões para o comportamento do armazenamento em cache HTTP 1.1 são encontrados em [RFC2616 seção 13.](#)) Estes cabeçalhos são:

- **ETag** (não personalizável): o cabeçalho da tag de entidade é um identificador exclusivo para a resposta enviada pelo MediaStore. CDNs e navegadores da web compatíveis com os padrões utilizam essa tag como chave para armazenar em cache o objeto. O MediaStore gera automaticamente um ETag para cada objeto quando ele é carregado. É possível [visualizar os detalhes de um objeto](#) para determinar o valor de sua ETag.
- **Last-Modified** (não personalizável): o valor desse cabeçalho indica a data e a hora em que o objeto foi modificado. O MediaStore gera automaticamente esse valor quando o objeto é carregado.
- **Cache-Control** (personalizável) – o valor deste cabeçalho controla por quanto tempo um objeto deve permanecer armazenado em cache antes de a CDN verificar se ele foi modificado. É possível definir esse cabeçalho como qualquer valor no upload de um objeto em um contêiner do MediaStore usando a [CLI](#) ou a [API](#). O conjunto completo de valores válidos é descrito na [documentação HTTP/1.1](#). Se você não definir esse valor ao fazer upload de um objeto, o MediaStore não retornará esse cabeçalho quando o objeto for recuperado.

Um caso de uso comum para o cabeçalho Cache-Control é a especificação de uma duração para armazenar o objeto em cache. Por exemplo, suponha que você tenha um arquivo de manifesto de vídeo que está sendo frequentemente substituído por um codificador. É possível definir a max-age como 10 para indicar que o objeto deve permanecer armazenado em cache por apenas 10 segundos. Ou suponha que você tenha um segmento de vídeo armazenado que nunca será substituído. É possível definir a max-age para esse objeto como 31536000 para armazená-lo em cache por cerca de 1 ano.

Solicitações condicionais

Solicitações condicionais ao MediaStore

O MediaStore responde de forma idêntica a solicitações condicionais (usando cabeçalhos de solicitação como `If-Modified-Since` e `If-None-Match`, conforme descrito em [RFC7232](#)) e solicitações incondicionais. Isso significa que quando o MediaStore recebe uma solicitação `GetObject` válida, o serviço sempre retorna o objeto, mesmo que o cliente já tenha o objeto.

Solicitações condicionais para CDNs

As CDNs que fornecem conteúdo em nome do podem processar solicitações condicionais retornando `304 Not Modified`, conforme descrito em [RFC7232 seção 4.1](#). Isso indica que não há necessidade de transferir o conteúdo completo do objeto, porque o solicitante já tem um objeto correspondente à solicitação condicional.

As CDNs (e outros caches compatíveis com HTTP/1.1) baseiam essas decisões nos cabeçalhos `ETag` e `Cache-Control` encaminhados pelos servidores de origem. Para controlar a frequência com que as CDNs consultam os servidores de origem do MediaStore por atualizações de objetos recuperados repetidamente, defina os cabeçalhos `Cache-Control` desses objetos ao fazer upload deles para o MediaStore.

Cotas no AWS Elemental MediaStore

O console de Service Quotas fornece informações sobre as cotas do AWS Elemental MediaStore. Além de visualizar as cotas padrão, você pode usar o console de Cotas de serviço para [solicitar aumentos de cota](#) para aquelas que podem ser ajustadas.

A tabela a seguir descreve cotas, anteriormente chamadas de limites, no AWS Elemental MediaStore. As cotas são o número máximo de recursos ou operações de serviço para sua conta da AWS.

Note

Para atribuir cotas a contêineres individuais em sua conta, entre em contato com o AWS Support ou com o gerente da conta. Essa opção pode ajudar a dividir os limites no nível da conta entre seus contêineres para evitar que um contêiner use toda a sua cota.

Recurso ou operação	Cota padrão	Comentários
Contêineres	100	O número máximo de contêineres que podem ser criados nesta conta.
Níveis de pasta	10	O número máximo de níveis de pasta que podem ser criados em um contêiner. É possível criar quantas pastas quiser, desde que não estejam aninhadas mais de 10 níveis em um contêiner.
Pastas	Ilimitado	É possível criar quantas pastas quiser, desde que não estejam aninhadas mais de 10 níveis em um contêiner.
Tamanho do objeto	25 MB	O tamanho de arquivo máximo de um único objeto.
Objetos	Ilimitado	Você pode fazer upload de quantos objetos quiser para uma pasta ou um contêiner em sua conta.

Recurso ou operação	Cota padrão	Comentários
Taxa de solicitações da API DeleteObject	100	O número máximo de solicitações de operação que você pode fazer por segundo. Solicitações adicionais são limitadas. É possível solicitar um aumento da cota .
Taxa de solicitações da API DescribeObject	1.000	O número máximo de solicitações de operação que você pode fazer por segundo. Solicitações adicionais são limitadas. É possível solicitar um aumento da cota .
Taxa de solicitações da API GetObject para disponibilidade-padrão de upload	1.000	O número máximo de solicitações de operação que você pode fazer por segundo. Solicitações adicionais são limitadas. É possível solicitar um aumento da cota .
Taxa de solicitações da API getObject para disponibilidade de upload de streaming	25	O número máximo de solicitações de operação que você pode fazer por segundo. Solicitações adicionais são limitadas. É possível solicitar um aumento da cota .
Taxa de solicitações da API ListItems	5	O número máximo de solicitações de operação que você pode fazer por segundo. Solicitações adicionais são limitadas. É possível solicitar um aumento da cota .

Recurso ou operação	Cota padrão	Comentários
Taxa de solicitações da API PutObject para codificação de transferência em blocos (também conhecida como disponibilidade de upload de streaming)	10	<p>O número máximo de solicitações de operação que você pode fazer por segundo. Solicitações adicionais são limitadas.</p> <p>É possível solicitar um aumento da cota. Na solicitação, especifique o TPS solicitado e o tamanho médio do objeto.</p>
Taxa de solicitações da API PutObject para disponibilidade-padrão de upload	100	<p>O número máximo de solicitações de operação que você pode fazer por segundo. Solicitações adicionais são limitadas.</p> <p>É possível solicitar um aumento da cota. Na solicitação, especifique o TPS solicitado e o tamanho médio do objeto.</p>
Regras em uma política de métrica	10	O número máximo de regras que podem ser incluídas em uma política de métrica.
Regras em uma política de ciclo de vida de objetos	10	O número máximo de regras que você pode incluir em uma política de ciclo de vida de objetos.

Informações relacionadas ao AWS Elemental MediaStore

A tabela a seguir lista os recursos relacionados que serão úteis à medida que você utilizar o AWS Elemental MediaStore.

- [Aulas e workshops](#) — Links para cursos de especialidades e baseados em perfil, bem como laboratórios autoguiados para ajudar a aperfeiçoar suas habilidades na AWS e a obter experiência prática.
- [Centro dos desenvolvedores da AWS](#) — Explore tutoriais, baixe ferramentas e informe-se sobre eventos para desenvolvedores da AWS.
- [Ferramentas do desenvolvedor da AWS](#) — Links para ferramentas de desenvolvedor, SDKs, toolkits de IDE e ferramentas da linha de comando para desenvolver e gerenciar aplicativos da AWS.
- [Centro de recursos de conceitos básicos](#) — Saiba como configurar a Conta da AWS, participar da comunidade da AWS e lançar seu primeiro aplicativo.
- [Tutoriais práticos](#) — Siga os tutoriais passo a passo para iniciar seu primeiro aplicativo na AWS.
- [Whitepapers da AWS](#) — Links para uma lista abrangente de whitepapers técnicos da AWS que abrangem tópicos como arquitetura, segurança e economia, elaborados pelos arquitetos de soluções da AWS ou por outros especialistas técnicos.
- [AWS Support Center](#): a central para criar e gerenciar seus casos do AWS Support. Também inclui links para outros recursos úteis, como fóruns, perguntas frequentes técnicas, status de integridade do serviço e AWS Trusted Advisor.
- [AWS Support](#) — A página Web principal para obter informações sobre o AWS Support, um canal de suporte de resposta rápida e com atendimento individual para ajudar a construir e a executar aplicativos na nuvem.
- [Entrar em contato](#) – Um ponto central de contato para consultas relativas a faturas da AWS, contas, eventos, uso abusivo e outros problemas.
- [Termos do site da AWS](#): informações detalhadas sobre nossos direitos autorais e marca registrada; sua conta, licença e acesso ao site, entre outros tópicos.

Histórico do documento para o guia do usuário

A tabela a seguir descreve a documentação desta versão do AWS Elemental MediaStore. Para receber notificações sobre atualizações dessa documentação, você pode se inscrever em um feed RSS.

Alteração	Descrição	Data
Melhoria no controle de acesso de origem (OAC)	Adicionadas informações sobre como usar o OAC com o AWS Elemental MediaStore.	17 de abril de 2023
Atualizações de cotas	Valor corrigido da cota e descrição para Rules in a Metric Policy.	25 de outubro de 2022
Campo ExpiresAt	Os registros de acesso agora incluem um campo ExpiresAt que indica a data e a hora de expiração do objeto com base nas regras de dados transitórios na política de ciclo de vida do contêiner.	16 de julho de 2020
Regras de transição do ciclo de vida	Agora é possível adicionar uma regra de transição do ciclo de vida à sua política de ciclo de vida do objeto que define objetos a serem movidos para a classe de armazenamento de acesso infrequente (IA) depois que eles atingem uma determinada idade.	20 de abril de 2020

[Contêiner vazio](#)

Agora você pode excluir todos os objetos dentro de um contêiner de uma vez.

7 de abril de 2020

[Suporte para métricas do Amazon CloudWatch](#)

É possível definir uma política de métrica para ditar quais métricas o MediaStore envia para o CloudWatch.

30 de março de 2020

[Curingas em regras de exclusão de objetos](#)

Em uma política de ciclo de vida de objeto, agora é possível usar um curinga em uma regra de exclusão de objetos. Isso permite especificar arquivos com base no nome de arquivo ou na extensão que deseja que o serviço exclua após um número de dias.

20 de dezembro de 2019

[Políticas de ciclo de vida de objetos](#)

Agora é possível adicionar uma regra à política de ciclo de vida do objeto que indica uma expiração por idade em segundos.

13 de setembro de 2019

[Suporte do AWS CloudFormation](#)

Agora, você pode usar um modelo do AWS CloudFormation para criar um contêiner automaticamente. O modelo do AWS CloudFormation gerencia os dados para cinco ações de API: criação de um contêiner, definição do registro de acesso, atualização da política de contêiner padrão, adição de uma política de CORS (compartilhamento de recursos entre origens) e adição de uma política de ciclo de vida de objetos.

17 de maio de 2019

[Cotas para disponibilidade de upload de streaming](#)

Para objetos com disponibilidade de upload de streaming (transferência de objetos em blocos), a operação `PutObject` não pode exceder 10 TPS e a operação `GetObject` não pode exceder 25 TPS.

8 de abril de 2019

[Transferência de objetos em blocos](#)

Adicionado suporte para transferência de objetos em blocos. Esse recurso permite especificar que um objeto está disponível para download antes de o objeto ser carregado completamente.

5 de abril de 2019

Registro de acesso	O AWS Elemental MediaStore agora é compatível com o registro de acesso em logs, que fornece registros detalhados das solicitações feitas para objetos em um contêiner.	25 de fevereiro de 2019
Políticas de ciclo de vida de objetos	Adicionado suporte para políticas de ciclo de vida de objeto, que controlam a data de expiração de objetos dentro do contêiner atual.	12 de dezembro de 2018
Aumentada a cota de tamanho do objeto	A cota para o tamanho de um objeto agora é de 25 MB.	10 de outubro de 2018
Aumentada a cota de tamanho do objeto	A cota para o tamanho de um objeto agora é de 20 MB.	6 de setembro de 2018
Integração do AWS CloudTrail	O conteúdo de integração do CloudTrail foi atualizado para se alinhar às alterações recentes no serviço do CloudTrail.	12 de julho de 2018
Colaboração da CDN	Adição de informações sobre como usar o AWS Elemental MediaStore com uma rede de entrega de conteúdo (CDN), como o Amazon CloudFront.	14 de abril de 2018

[Configurações do CORS](#)

Agora, o AWS Elemental MediaStore oferece suporte ao compartilhamento de recursos de origem cruzada (CORS), que permite que as aplicações web do cliente de um domínio interajam com recursos em outro domínio.

7 de fevereiro de 2018

[Novo guia e serviço](#)

Esta é a versão inicial do serviço de originação e armazenamento de vídeo, o AWS Elemental MediaStore e do Guia do usuário do AWS Elemental MediaStore.

27 de novembro de 2017

Note

- Os Serviços de Mídia da AWS não foram projetados nem devem ser usados com aplicativos ou em situações que exijam desempenho à prova de falhas, como operações de segurança da vida, sistemas de navegação ou comunicação, controle de tráfego aéreo ou máquinas de suporte à vida em que a indisponibilidade, a interrupção ou a falha dos serviços pode levar à morte, a danos corporais, danos materiais ou danos ambientais.

Glossário do AWS

Para obter a terminologia mais recente da AWS, consulte o [glossário da AWS](#) na Referência do Glossário da AWS.