aws

Manual do usuário

Amazon ECR

Versão da API 2015-09-21

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon ECR: Manual do usuário

Copyright © 2024 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 os clientes ou que deprecie ou desprestigie a Amazon. Todas as outras marcas comerciais que não são propriedade da Amazon pertencem aos respectivos proprietários, os quais podem ou não ser afiliados, estar conectados ou ser patrocinados pela Amazon.

Table of Contents

O que é o Amazon Elastic Container Registry?

O Amazon Elastic Container Registry (Amazon ECR) é AWS um serviço gerenciado de registro de imagens de contêineres que é seguro, escalável e confiável. O Amazon ECR oferece suporte a repositórios privados com permissões baseadas em recursos usando o IAM. AWS Isso é para que usuários ou instâncias do Amazon EC2 especificados possam acessar seus repositórios e imagens de contêiner. É possível usar a CLI preferida para enviar, extrair e gerenciar imagens do Docker, imagens da Open Container Initiative (OCI)e artefatos compatíveis com OCI.

> **Note**
>
> O Amazon ECR também suporta repositórios de imagens de contêiner público. Para obter mais informações, consulte O que é o Amazon ECR Public no Manual do usuário do Amazon ECR Public.

A equipe de serviços de AWS contêineres mantém um roteiro público em. GitHub Ele contém informações sobre o que as equipes estão trabalhando e permite que todos os AWS clientes forneçam feedback direto. Para obter mais informações, consulte Roteiro de contêineres da AWS.

Componentes do Amazon ECR

O Amazon ECR tem os seguintes componentes:

Registro

Um registro privado do Amazon ECR é fornecido para cada AWS conta; você pode criar um ou mais repositórios em seu registro e armazenar imagens do Docker, imagens da Open Container Initiative (OCI) e artefatos compatíveis com OCI neles. Para ter mais informações, consulte Registro privado do Amazon ECR.

Token de autorização

Seu cliente deve autenticar-se em um registro privado do Amazon ECR como um usuário AWS antes de poder enviar e receber imagens. Para ter mais informações, consulte Autenticação de registro privado no Amazon ECR.

Repositório

Um repositório do Amazon ECR contém suas imagens do Docker, imagens da Open Container Initiative (OCI) e artefatos compatíveis com OCI. Para ter mais informações, consulte Repositórios privados do Amazon ECR.

Política de repositório

Você pode controlar o acesso aos repositórios e ao conteúdo contido neles com as políticas de repositório. Para ter mais informações, consulte Políticas de repositório privado no Amazon ECR.

Imagem

É possível enviar e extrair imagens de contêiner dos seus repositórios. Você pode usar essas imagens localmente no seu sistema de desenvolvimento ou nas definições de tarefas do Amazon ECS e especificações de pod do Amazon EKS. Para obter mais informações, consulte Uso de imagens do Amazon ECR com o Amazon ECS e Uso de imagens do Amazon ECR com o Amazon EKS.

Recursos do Amazon ECR

O Amazon ECR fornece os seguintes recursos:

- As políticas de ciclo de vida ajudam a gerenciar o ciclo de vida das imagens em seus repositórios. Você define regras que resultam na limpeza das imagens não utilizadas. Você pode testar as regras antes de aplicá-las ao repositório. Para ter mais informações, consulte Automatize a limpeza de imagens usando políticas de ciclo de vida no Amazon ECR.
- A verificação de imagens ajuda a identificar vulnerabilidades de software nas imagens de seu contêiner. Cada repositório pode ser configurado para verifica no envio. Isso garante que cada nova imagem enviada para o repositório seja verificada. Em seguida, você pode recuperar os resultados da verificação das imagens. Para ter mais informações, consulte Digitalize imagens em busca de vulnerabilidades de software no Amazon ECR.
- A replicação entre regiões e entre contas torna mais fácil para você ter suas imagens onde precisa. Isso é definido como uma configuração do registro e é feito por região. Para ter mais informações, consulte Configurações de registro privado no Amazon ECR.
- As regras de cache de pull through fornecem uma maneira de armazenar em cache repositórios em um registro upstream no seu registro privado do Amazon ECR. Usando uma regra de cache pull through, o Amazon ECR entrará em contato com o registro upstream periodicamente para garantir que a imagem armazenada em cache no seu registro privado do Amazon ECR esteja

atualizada. Para ter mais informações, consulte Sincronize um registro upstream com um registro privado do Amazon ECR.

Como começar a usar o Amazon ECR

Se você estiver usando o Amazon Elastic Container Service (Amazon ECS) ou o Amazon Elastic Kubernetes Service (Amazon EKS), observe que a configuração desses dois serviços é semelhante à configuração do Amazon ECR porque o Amazon ECR é uma extensão dos dois serviços.

Ao usar o AWS Command Line Interface com o Amazon ECR, use uma versão do AWS CLI que suporte os recursos mais recentes do Amazon ECR. Se você não encontrar suporte para um recurso do Amazon ECR no AWS CLI, atualize para a versão mais recente do AWS CLI. Para obter informações sobre como instalar a versão mais recente do AWS CLI, consulte Instalar ou atualizar para a versão mais recente do AWS CLI no Guia do AWS Command Line Interface Usuário.

Para saber como enviar uma imagem de contêiner para um repositório privado do Amazon ECR usando o AWS CLI e o Docker, consulte. Movendo uma imagem ao longo de seu ciclo de vida no Amazon ECR

Preços do Amazon ECR

Com o Amazon ECR, você paga apenas pela quantidade de dados armazenada em seus repositórios e pela transferência de dados nos envios e extrações de imagens. Para obter mais informações, consulte a Definição de preço do Amazon ECR.

Movendo uma imagem ao longo de seu ciclo de vida no Amazon ECR

Se você estiver usando o Amazon ECR pela primeira vez, use as etapas a seguir com a CLI do Docker e a AWS CLI para criar uma imagem de amostra, autenticar-se no registro padrão e criar um repositório privado. Em seguida, envie uma imagem e extraia uma imagem do repositório privado. Quando você terminar de usar a imagem de amostra, exclua a imagem de amostra e o repositório.

Para usar o AWS Management Console em vez do AWS CLI, consultethe section called “Criação de um repositório para armazenar imagens”.

Para obter mais informações sobre as outras ferramentas disponíveis para gerenciar seus AWS recursos, incluindo os diferentes AWS SDKs, kits de ferramentas do IDE e ferramentas de linha de PowerShell comando do Windows, consulte http://aws.amazon.com/tools/.

Pré-requisitos

Se você não tiver o Docker mais recente AWS CLI instalado e pronto para uso, use as etapas a seguir para instalar essas duas ferramentas.

Instale o AWS CLI

Para usar o AWS CLI com o Amazon ECR, instale a AWS CLI versão mais recente. Para obter informações, consulte Instalar a AWS Command Line Interface no Guia do usuário da AWS Command Line Interface .

Instalar o Docker

O Docker está disponível em muitos sistemas operacionais diferentes, incluindo a maioria das distribuições modernas do Linux, como o Ubuntu, e até no MacOS e no Windows. Para obter mais informações sobre como instalar o Docker no seu sistema operacional, consulte o Guia de instalação do Docker.

Não é necessário um sistema de desenvolvimento local para usar o Docker. Se você já usa o Amazon EC2, pode iniciar uma instância do Amazon Linux 2023 e instalar o Docker para começar.

Se você já tiver um Docker instalado, vá para Etapa 1: criar uma imagem do Docker.

Para instalar o Docker em uma instância do Amazon EC2 usando uma AMI do Amazon Linux 2023

1. Inicie uma instância com a mais recente AMI do Amazon Linux 2023. Para obter mais informações, consulte Lançamento de uma instância no Guia do usuário do Amazon EC2.
2. Conecte-se à sua instância. Para obter mais informações, consulte Connect to Your Linux Instance no Guia do usuário do Amazon EC2.
3. Atualize os pacotes instalados e o cache de pacotes em sua instância.

```
sudo yum update -y
```

4. Instale o pacote do Docker Community Edition mais recente.

```
sudo yum install docker
```

5. Inicie o serviço Docker.

```
sudo service docker start
```

6. Adicione o `ec2-user` ao grupo `docker`, de modo que você possa executar comandos do Docker sem usar o `sudo`.

```
sudo usermod -a -G docker ec2-user
```

7. Faça logout e login novamente para selecionar as novas permissões do grupo `docker`. É possível fazer isso ao fechar a janela de terminal SSH atual e se reconectar à sua instância em outra janela. Sua nova sessão SSH terá as permissões de grupo `docker` apropriadas.
8. Verifique se o `ec2-user` pode executar comandos do Docker sem `sudo`.

```
docker info
```

> **Note**
>
> Em alguns casos, pode ser necessário reinicializar sua instância para fornecer permissões para o `ec2-user` acessar o daemon do Docker. Tente reinicializar sua instância se você vir o seguinte erro:

```
Cannot connect to the Docker daemon. Is the docker daemon running on this
 host?
```

Etapa 1: criar uma imagem do Docker

Nesta etapa, crie uma imagem do Docker de uma aplicação Web simples e teste-a no sistema ou na instância do Amazon EC2 local.

Para criar uma imagem do Docker de um aplicativo web simples

1. Crie um arquivo chamado `Dockerfile`. Um Dockerfile é um manifesto que descreve a imagem básica a ser usada para a sua imagem do Docker e o que você deseja instalar e executar nela. Para obter mais informações sobre a Dockerfiles, visite Referência de Dockerfiles.

   ```
   touch Dockerfile
   ```

2. Edite o `Dockerfile` que você acabou de criar e adicione o conteúdo a seguir.

   ```
   FROM public.ecr.aws/amazonlinux/amazonlinux:latest

   # Install dependencies
   RUN yum update -y && \
    yum install -y httpd

   # Install apache and write hello world message
   RUN echo 'Hello World!' > /var/www/html/index.html

   # Configure apache
   RUN echo 'mkdir -p /var/run/httpd' >> /root/run_apache.sh && \
    echo 'mkdir -p /var/lock/httpd' >> /root/run_apache.sh && \
    echo '/usr/sbin/httpd -D FOREGROUND' >> /root/run_apache.sh && \
    chmod 755 /root/run_apache.sh

   EXPOSE 80

   CMD /root/run_apache.sh
   ```

Esse Dockerfile usa a imagem pública do Amazon Linux 2 hospedada no Amazon ECR Public. As instruções `RUN` atualizam os caches de pacotes, instalam alguns pacotes de software para o servidor Web e, em seguida, gravam o conteúdo de “Hello World!” na raiz do documento dos servidores Web. A instrução `EXPOSE` expõe a porta 80 do contêiner e a instrução `CMD` inicia o servidor Web.

3. Crie a imagem do Docker do seu Dockerfile.

> **Note**
>
> Algumas versões do Docker podem exigir o caminho completo para o seu Dockerfile no seguinte comando, em vez de o caminho relativo mostrado abaixo.

```
docker build -t hello-world .
```

4. Liste a sua imagem do contêiner.

```
docker images --filter reference=hello-world
```

Saída:

```
REPOSITORY          TAG                 IMAGE ID            CREATED
 SIZE
hello-world         latest              e9ffedc8c286        4 minutes ago
 194MB
```

5. Execute a imagem recém-criada. A opção `-p 80:80` mapeia a porta 80 exposta no contêiner para a porta 80 no sistema de host. Para obter mais informações sobre o docker run, acesse a Referência de execução do Docker.

```
docker run -t -i -p 80:80 hello-world
```

> **Note**
>
> A saída do servidor Web Apache é exibida na janela do terminal. É possível ignorar a mensagem "`Could not reliably determine the fully qualified domain name`".

6. Abra um navegador e aponte para o servidor que está executando o Docker e hospedando seu contêiner.

   - Se você estiver usando uma instância do EC2, esse é o valor Public DNS para o servidor, que é o mesmo endereço usado para se conectar à instância com o SSH. Certifique-se de que o security group para sua instância permita o tráfego de entrada na porta 80.
   - Se você estiver executando o Docker localmente, aponte seu navegador para http://localhost/.
   - Se você estiver usando docker-machine em um computador Windows ou Mac, encontre o endereço IP da VirtualBox VM que está hospedando o Docker com o docker-machine ip comando, substituindo *`machine-name pelo nome`* da máquina docker que você está usando.

     ```
     docker-machine ip machine-name
     ```

   Você deve ver uma página da Web com seu “Hello, World!” instrução.

7. Interrompa o contêiner do Docker digitando Ctrl+c.

Etapa 2: autenticar-se no registro padrão

Depois de instalar e configurar o AWS CLI, autentique a CLI do Docker em seu registro padrão. Desta forma, o comando docker pode enviar e extrair imagens com o Amazon ECR. O AWS CLI fornece um get-login-password comando para simplificar o processo de autenticação.

Para autenticar o Docker em um registro do Amazon ECR com get-login-password, execute o comando. aws ecr get-login-password Ao transmitir o token de autenticação para o comando docker login, use o valor `AWS` para o nome de usuário, e especifique o URI de registro do Amazon ECR para o qual deseja fazer a autenticação. Se autenticar em vários registros, você deverá repetir o comando para cada registro.

> **Important**
>
> Se você receber um erro, instale ou atualize para a versão mais recente da AWS CLI. Para obter mais informações, consulte Installing the AWS Command Line Interface (Instalar a AWS Command Line Interface) no User Guide (Guia do usuário da).

- get-login-password (AWS CLI)

```
aws ecr get-login-password --region region | docker login --username AWS --password-stdin aws_account_id.dkr.ecr.region.amazonaws.com
```

- Obtenha ECR () LoginCommandAWS Tools for Windows PowerShell

```
(Get-ECRLoginCommand).Password | docker login --username AWS --password-stdin aws_account_id.dkr.ecr.region.amazonaws.com
```

Etapa 3: criar um repositório

Agora que tem uma imagem para enviar ao Amazon ECR, você precisa criar um repositório para guardá-la. Neste exemplo, você cria um repositório chamado `hello-repository` para o qual enviará a imagem `hello-world:latest` posteriormente. Para criar um repositório, execute o seguinte comando:

```
aws ecr create-repository \
    --repository-name hello-repository \
    --region region
```

Etapa 4: enviar uma imagem ao Amazon ECR

Agora você pode enviar a imagem ao repositório do Amazon ECR que criou na seção anterior. Use a docker CLI para enviar imagens depois que os seguintes pré-requisitos forem atendidos:

- A versão mínima do docker está instalada: 1.7.
- O token de autorização do Amazon ECR foi configurado comdocker login.
- O repositório do Amazon ECR existe, e o usuário tem acesso para enviar imagens ao repositório.

Depois que esses pré-requisitos forem atendidos, você poderá enviar a imagem ao repositório recém-criado no registro padrão da sua conta.

Para marcar e enviar uma imagem para o Amazon ECR

1. Liste as imagens que você armazenou localmente para identificar a imagem a ser marcada e enviada.

   ```
   docker images
   ```

   Saída:

   ```
   REPOSITORY          TAG                 IMAGE ID            CREATED
    VIRTUAL SIZE
   hello-world         latest              e9ffedc8c286        4 minutes ago
    241MB
   ```

2. Marque a imagem a ser enviada ao seu repositório.

   ```
   docker tag hello-world:latest aws_account_id.dkr.ecr.region.amazonaws.com/hello-
   repository
   ```

3. Envie a imagem.

   ```
   docker push aws_account_id.dkr.ecr.region.amazonaws.com/hello-repository
   ```

   Saída:

   ```
   The push refers to a repository [aws_account_id.dkr.ecr.region.amazonaws.com/hello-
   repository] (len: 1)
   e9ae3c220b23: Pushed
   a6785352b25c: Pushed
   0998bf8fb9e9: Pushed
   0a85502c06c9: Pushed
   latest: digest: sha256:215d7e4121b30157d8839e81c4e0912606fca105775bb0636EXAMPLE
    size: 6774
   ```

Etapa 5: extrair uma imagem do Amazon ECR

Depois que sua imagem for enviada para o repositório Amazon ECR, você poderá extraí-la de outros locais. Use a docker CLI para extrair imagens depois que os seguintes pré-requisitos forem atendidos:

- A versão mínima do docker está instalada: 1.7.
- O token de autorização do Amazon ECR foi configurado comdocker login.
- O repositório do Amazon ECR existe, e o usuário tem acesso para extrair imagens do repositório.

Depois que esses pré-requisitos forem atendidos, você poderá extrair a imagem. Para extrair a imagem de exemplo do Amazon ECR, execute o seguinte comando:

```
docker pull aws_account_id.dkr.ecr.region.amazonaws.com/hello-repository:latest
```

Saída:

```
latest: Pulling from hello-repository
0a85502c06c9: Pull complete
0998bf8fb9e9: Pull complete
a6785352b25c: Pull complete
e9ae3c220b23: Pull complete
Digest: sha256:215d7e4121b30157d8839e81c4e0912606fca105775bb0636EXAMPLE
Status: Downloaded newer image for aws_account_id.dkr.region.amazonaws.com/hello-
repository:latest
```

Etapa 6: excluir uma imagem

Se você não precisar mais de uma imagem em um dos seus repositórios, poderá excluí-la. Para excluir uma imagem, especifique o repositório em que ela está e um `imageDigest` valor `imageTag` ou para a imagem. O exemplo a seguir exclui uma imagem no `hello-repository` repositório com a tag de imagem. `latest` Para excluir sua imagem de exemplo do repositório, execute o seguinte comando:

```
aws ecr batch-delete-image \
      --repository-name hello-repository \
      --image-ids imageTag=latest \
```

```
      --region region
```

Etapa 7: excluir um repositório

Se você não precisar mais de um repositório inteiro de imagens, poderá excluir o repositório. O exemplo a seguir usa o `--force` sinalizador para excluir um repositório que contém imagens. Para excluir um repositório que contém imagens (e todas as imagens contidas nele), execute o seguinte comando:

```
aws ecr delete-repository \
      --repository-name hello-repository \
      --force \
      --region region
```

Otimização do performance para o Amazon ECR

Você pode usar as seguintes recomendações sobre configurações e estratégias para otimizar o desempenho ao usar o Amazon ECR.

Use o Docker 1.10 e versões posteriores para utilizar os uploads simultâneos da camada

As imagens de Docker são compostas por camadas, que são estágios de compilação intermediários da imagem. Cada linha em um Dockerfile resulta na criação de uma nova camada. Quando você usa o Docker 1.10 e versões posteriores, o Docker envia por padrão o maior número possível de camadas como carregamentos simultâneos ao Amazon ECR, o que resulta em tempos de carregamento mais rápidos.

Use uma imagem de base menor

As imagens padrão disponíveis por meio do Docker Hub podem conter muitas dependências das quais seu aplicativo não precisa. Considere o uso de uma imagem menor criada e mantida por outras pessoas da comunidade do Docker ou compile sua própria imagem de base usando a imagem mínima de scratch do Docker. Para obter mais informações, consulte Criar uma imagem de base na documentação do Docker.

Coloque as dependências que mudam menos no início do Dockerfile

O Docker armazena as camadas em cache, o que acelera os tempos de compilação. Se nada tiver sido alterado na camada desde a última compilação, o Docker usará a versão armazenada em cache, em vez de compilar a camada novamente. No entanto, cada camada depende das camadas que vieram antes dela. Se uma camada mudar, o Docker a compilará novamente, bem como todas as camadas que vierem depois dela.

Para minimizar o tempo necessário para compilar um arquivo de Dockerfile e fazer upload das camadas novamente, considere colocar as dependências que mudam com menos frequência no início do Dockerfile. E, aquelas que mudam rapidamente, (como o código-fonte do seu aplicativo) mais à frente na pilha.

Encadeie os comandos para evitar o armazenamento desnecessário de arquivos

Os arquivos intermediários criados em uma camada continuarão fazendo parte dela, mesmo que sejam excluídos em uma camada subsequente. Considere o seguinte exemplo:

```
WORKDIR /tmp
RUN wget http://example.com/software.tar.gz
```

```
RUN wget tar -xvf software.tar.gz
RUN mv software/binary /opt/bin/myapp
RUN rm software.tar.gz
```

Neste exemplo, as camadas criadas pelo primeiro e pelo segundo comando EXECUTAR contêm o arquivo original .tar.gz e todos os seus conteúdos descompactados. Embora o arquivo .tar.gz seja excluído pelo quarto comando EXECUTAR. Esses comandos podem ser encadeados em uma única instrução EXECUTAR para garantir que esses arquivos desnecessários não façam parte da imagem de Docker final:

```
WORKDIR /tmp
RUN wget http://example.com/software.tar.gz &&\
    wget tar -xvf software.tar.gz &&\
    mv software/binary /opt/bin/myapp &&\
    rm software.tar.gz
```

Use o endpoint regional mais próximo

Você pode reduzir a latência na extração de imagens do Amazon ECR usando o endpoint regional mais próximo de onde seu aplicativo está sendo executado. Se seu aplicativo estiver sendo executado em uma instância do Amazon EC2, você pode usar o seguinte código de shell para obter a região da zona de disponibilidade da instância:

```
REGION=$(curl -s http://169.254.169.254/latest/meta-data/placement/availability-zone
 |\
    sed -n 's/\(\d*\)[a-zA-Z]*$/\1/p')
```

A região pode ser passada para AWS CLI comandos usando o --region parâmetro ou definida como a região padrão para um perfil usando o aws configure comando. Você também pode definir a região ao fazer chamadas usando o AWS SDK. Para obter mais informações, consulte a documentação do SDK para a sua linguagem de programação específica.

Registro privado do Amazon ECR

Um registro privado do Amazon ECR hospeda as imagens de contêiner em uma arquitetura altamente disponível e escalável. É possível usar o seu registro privado para gerenciar repositórios de imagens privados que consistem em imagens do Docker e da Open Container Initiative (OCI). Cada conta da AWS é fornecida com um registro privado padrão do Amazon ECR. Para obter mais informações sobre registros públicos do Amazon ECR, consulte Registros públicos no Manual do usuário do Amazon Elastic Container Registry.

Conceitos do registro privado

- O URL do seu registro privado padrão é `https://aws_account_id.dkr.ecr.us-west-2.amazonaws.com`.
- Por padrão, sua conta tem acesso de leitura e gravação aos repositórios no seu registro privado padrão. No entanto, os usuários precisam de permissões para fazer chamadas para as APIs do Amazon ECR e enviar ou extrair imagens de e para seus repositórios privados. O Amazon ECR fornece várias políticas gerenciadas para controlar o acesso do usuário em diversos níveis. Para ter mais informações, consulte Exemplos de políticas baseadas em identidade do Amazon Elastic Container Registry.
- Você deve autenticar seu cliente do Docker em um registro privado para poder usar os comandos docker push e docker pull para enviar para, e extrair imagens dos, repositórios nesse registro. Para obter mais informações, consulte Autenticação de registro privado no Amazon ECR.
- Os repositórios privados podem ser controlados com políticas de acesso de usuários do e políticas de repositório. Para obter mais informações sobre políticas de repositórios, consulte Políticas de repositório privado no Amazon ECR.
- Os repositórios em seu registro privado podem ser replicados entre regiões em seu próprio registro privado e entre contas separadas, configurando a replicação para seu registro privado. Para ter mais informações, consulte Replicação de imagens privadas no Amazon ECR.

Autenticação de registro privado no Amazon ECR

Você pode usar o AWS Management Console AWS CLI, o ou os AWS SDKs para criar e gerenciar repositórios privados. Você também pode usar esses métodos para realizar algumas ações em imagens, como listá-las ou excluí-las. Esses clientes usam métodos de AWS autenticação padrão.

Embora seja possível usar a API do Amazon ECR para enviar e extrair imagens, é muito mais provável que você use a CLI do Docker ou uma biblioteca do Docker específica para a linguagem.

A CLI do Docker não suporta métodos de autenticação nativos do IAM. É necessário realizar etapas adicionais para que o Amazon ECR possa autenticar e autorizar solicitações de extração e envio do Docker.

Os métodos de autenticação de registro a seguir estão detalhados nas seções a seguir estão disponíveis.

Uso de auxiliar de credenciais do Amazon ECR

O Amazon ECR fornece um auxiliar de credenciais do Docker que facilita o armazenamento e o uso de credenciais do Docker ao enviar e extrair imagens do Amazon ECR. Para obter as etapas de instalação e configuração, consulte Auxiliar de credenciais do Docker do Amazon ECR.

> **Note**
>
> No momento, o auxiliar de credencial do Amazon ECR Docker não oferece suporte a autenticação multifator (MFA).

Uso de um token de autorização

O escopo de permissão de um token de autorização corresponde ao do principal do IAM usado para recuperar o token de autenticação. Um token de autenticação é usado para acessar qualquer registro do Amazon ECR ao qual o principal do IAM tenha acesso e é válido por 12 horas. Para obter um token de autorização, você deve usar a operação da GetAuthorizationTokenAPI para recuperar um token de autorização codificado em base64 contendo o nome de usuário `AWS` e uma senha codificada. O AWS CLI `get-login-password` comando simplifica isso recuperando e decodificando o token de autorização, que você pode então canalizar para um docker login comando para autenticar.

Para autenticar o Docker para um registro privado do Amazon ECR com get-login

- Para autenticar o Docker em um registro do Amazon ECR com get-login-password, execute o comando. aws ecr get-login-password Ao transmitir o token de autenticação para o comando docker login, use o valor `AWS` para o nome de usuário, e especifique o URI de registro do

Amazon ECR para o qual deseja fazer a autenticação. Se autenticar em vários registros, você deverá repetir o comando para cada registro.

> **Important**
>
> Se você receber um erro, instale ou atualize para a versão mais recente da AWS CLI. Para obter mais informações, consulte Installing the AWS Command Line Interface (Instalar a AWS Command Line Interface) no User Guide (Guia do usuário da).

- get-login-password (AWS CLI)

```
aws ecr get-login-password --region region | docker login --username AWS --
password-stdin aws_account_id.dkr.ecr.region.amazonaws.com
```

- Obtenha ECR () LoginCommandAWS Tools for Windows PowerShell

```
(Get-ECRLoginCommand).Password | docker login --username AWS --password-
stdin aws_account_id.dkr.ecr.region.amazonaws.com
```

Uso da autenticação de API HTTP

O Amazon ECR suporta API HTTP de registro do Docker. No entanto, como o Amazon ECR é um registro privado, você deve fornecer um token de autorização com cada solicitação HTTP. Você pode adicionar um cabeçalho de autorização HTTP usando a `-H` opção for curl e passar o token de autorização fornecido pelo get-authorization-token AWS CLI comando.

Para autenticar com a API HTTP do Amazon ECR

1. Recupere um token de autorização com o AWS CLI e defina-o como uma variável de ambiente.

```
TOKEN=$(aws ecr get-authorization-token --output text --query
 'authorizationData[].authorizationToken')
```

2. A fim de fazer a autenticação para a API, passe a variável `$TOKEN` para a opção `-H` de curl. Por exemplo, o comando a seguir lista as tags da imagem em um repositório do Amazon ECR. Para obter mais informações, consulte a documentação de referência da API HTTP de registro do Docker.

```
curl -i -H "Authorization: Basic $TOKEN"
 https://aws_account_id.dkr.ecr.region.amazonaws.com/v2/amazonlinux/tags/list
```

A saída é a seguinte:

```
HTTP/1.1 200 OK
Content-Type: text/plain; charset=utf-8
Date: Thu, 04 Jan 2018 16:06:59 GMT
Docker-Distribution-Api-Version: registry/2.0
Content-Length: 50
Connection: keep-alive

{"name":"amazonlinux","tags":["2017.09","latest"]}
```

Configurações de registro privado no Amazon ECR

O Amazon ECR usa configurações do registro privado para configurar recursos no nível do registro. As configurações de registro privado são definidas separadamente para cada região. Você pode usar as configurações do registro privado para configurar os seguintes recursos.

- Permissões de registro - uma política de permissões de registro fornece controle sobre as permissões de replicação e extração de cache. Para ter mais informações, consulte Permissões de registro privado no Amazon ECR.
- Regras de cache de pull-through - você pode criar regras de cache de pull-through para armazenar em cache imagens de um registro público externo em seu registro privado do Amazon ECR. Para ter mais informações, consulte Sincronize um registro upstream com um registro privado do Amazon ECR.
- Configuração de replicação - A configuração de replicação é usada para controlar se seus repositórios são copiados entre regiões e contas do AWS . Para mais informações, consulte Replicação de imagens privadas no Amazon ECR.
- Modelos de criação de repositório - Um modelo de criação de repositório é usado para definir as configurações padrão a serem aplicadas quando novos repositórios são criados pelo Amazon ECR em seu nome. Por exemplo, repositórios criados por uma ação de cache de pull-through. Para ter mais informações, consulte Modelos para controlar repositórios criados durante uma ação de extração do cache.

- Configuração de verificação: por padrão, seu registro está habilitado para verificação básica. Você pode habilitar a verificação avançada que fornece um modo de verificação automatizado e contínuo que verifica as vulnerabilidades do sistema operacional e do pacote da linguagem de programação. Para ter mais informações, consulte Digitalize imagens em busca de vulnerabilidades de software no Amazon ECR.

Permissões de registro privado no Amazon ECR

O Amazon ECR usa uma política de registro para conceder permissões a uma entidade principal do AWS no nível de registro privado. Essas permissões são usadas para definir o escopo de acesso à replicação e recursos de pull through do cache.

O Amazon ECR só impõe as seguintes permissões no nível de registro privado. Se alguma ação adicional for adicionada à política de registro, ocorrerá um erro.

- `ecr:ReplicateImage` — Concede permissão a outra conta, chamada de registro de origem, para replicar suas imagens para o registro. Isso é usado apenas para replicação entre contas.
- `ecr:BatchImportUpstreamImage`: concede permissão para recuperar a imagem externa e importá-la para o registro privado.
- `ecr:CreateRepository`: concede permissão para criar um repositório em um registro privado. Essa permissão é necessária se o armazenamento do repositório estiver replicado ou as imagens em cache ainda não existirem no registro privado.

> **Note**
>
> Embora seja possível adicionar a ação `ecr:*` a uma política de permissões de registro privado, é considerada uma prática recomendada adicionar apenas as ações específicas necessárias com base no recurso que você está usando, em vez de usar um curinga.

Tópicos

- Exemplos de políticas de registro privado para o Amazon ECR
- Conceder permissões de registro para replicação entre contas no Amazon ECR
- Conceder permissões de registro para extrair o cache no Amazon ECR

Exemplos de políticas de registro privado para o Amazon ECR

Os exemplos a seguir mostram instruções de políticas de registro que você pode usar para controlar as permissões que os usuários têm em seu registro do Amazon ECR.

> **Note**
>
> Em cada exemplo, se a ação `ecr:CreateRepository` for removida da instrução de permissão do registro, a replicação ainda pode ocorrer. No entanto, para uma replicação ser bem-sucedida, você precisa criar repositórios com o mesmo nome em sua conta.

Exemplo: permitir que o usuário raiz de uma conta de origem replique todos os repositórios

A política de permissões de registro a seguir permite que o usuário raiz de uma conta de origem replique todos os repositórios.

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"ReplicationAccessCrossAccount",
            "Effect":"Allow",
            "Principal":{
                "AWS":"arn:aws:iam::source_account_id:root"
            },
            "Action":[
                "ecr:CreateRepository",
                "ecr:ReplicateImage"
            ],
            "Resource": [
                "arn:aws:ecr:us-west-2:your_account_id:repository/*"
            ]
        }
    ]
}
```

Exemplo: permitir usuários root de várias contas

A política de permissões de registro a seguir tem duas declarações. Cada instrução permite que o usuário raiz de uma conta de origem replique todos os repositórios.

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"ReplicationAccessCrossAccount",
            "Effect":"Allow",
            "Principal":{
                "AWS":"arn:aws:iam::source_account_id:root"
            },
            "Action":[
                "ecr:CreateRepository",
                "ecr:ReplicateImage"
            ],
            "Resource": [
                "arn:aws:ecr:us-west-2:your_account_id:repository/*"
            ]
        },
        {
            "Sid":"ReplicationAccessCrossAccount",
            "Effect":"Allow",
            "Principal":{
                "AWS":"arn:aws:iam::source_account_id:root"
            },
            "Action":[
                "ecr:CreateRepository",
                "ecr:ReplicateImage"
            ],
            "Resource": [
                "arn:aws:ecr:us-west-2:your_account_id:repository/*"
            ]
        }
    ]
}
```

Exemplo: permitir que o usuário raiz de uma conta de origem replique todos os repositórios com o prefixo **`prod-`**.

A política de permissões de registro a seguir permite que o usuário raiz de uma conta de origem replique todos os repositórios que começam com. `prod-`

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"ReplicationAccessCrossAccount",
            "Effect":"Allow",
            "Principal":{
                "AWS":"arn:aws:iam::source_account_id:root"
            },
            "Action":[
                "ecr:CreateRepository",
                "ecr:ReplicateImage"
            ],
            "Resource": [
                "arn:aws:ecr:us-west-2:your_account_id:repository/prod-*"
            ]
        }
    ]
}
```

Conceder permissões de registro para replicação entre contas no Amazon ECR

O tipo de política entre contas é usado para conceder permissões a uma entidade principal do AWS , permitindo a replicação dos repositórios de um registro de origem para o seu registro. Por padrão, você tem permissão para configurar a replicação entre regiões no seu próprio registro. Só é necessário configurar a política de registro se estiver concendendo outra permissão de conta para replicar conteúdo para o seu registro.

Uma política de registro deve conceder permissão para a ação de API `ecr:ReplicateImage`. Essa API é uma API interna do Amazon ECR que pode replicar imagens entre regiões ou contas. Você também pode conceder a permissão `ecr:CreateRepository`, que permite que o Amazon ECR crie repositórios em seu registro se eles ainda não existirem. Se a permissão `ecr:CreateRepository` não for fornecida, um repositório com o mesmo nome que o repositório

de origem deve ser criado manualmente no seu registro. Se nenhuma das duas alternativas foi realizada, a replicação falha. Qualquer falha CreateRepository ou ação ReplicateImage da API aparece no CloudTrail.

Para configurar uma política de permissões para replicação (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/.
2. Na barra de navegação, escolha a região para configurar a sua política de registro.
3. No painel de navegação, escolha Private registry (Registro privado), Registry permissions (Permissões do registro).
4. Na página Registry permissions (Permissões do registro), escolha Generate statement (Gerar declaração).
5. Realize as seguintes etapas para definir a instrução da política usando o gerador de políticas.
   a. Em Policy type (Tipo de Política), escolha Cross-account policy (Política entre contas).
   b. Para Statement ID (ID da instrução), insira um ID de instrução exclusivo. Este campo é usado como o `Sid` na política de registro.
   c. Para Accounts (Contas), insira os IDs de conta para cada conta à qual você deseja conceder permissões. Ao especificar vários IDs de conta, separe-os com uma vírgula.
6. Expanda a seção Preview policy statement (Previsualizar instrução da política para rever a instrução da política de permissões do registro.
7. Depois que a instrução da política for confirmada, escolha Add to policy (Adicionar à política) para salvar a política em seu registro.

Para configurar uma política de permissões para replicação (AWS CLI)

1. Crie um arquivo denominado `registry_policy.json` e preencha-o com uma política de registro.

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"ReplicationAccessCrossAccount",
            "Effect":"Allow",
            "Principal":{
                "AWS":"arn:aws:iam::source_account_id:root"
```

```
            },
            "Action":[
                "ecr:CreateRepository",
                "ecr:ReplicateImage"
            ],
            "Resource": [
                "arn:aws:ecr:us-west-2:your_account_id:repository/*"
            ]
        }
    ]
}
```

2. Crie a política de registro usando o arquivo de política.

```
aws ecr put-registry-policy \
      --policy-text file://registry_policy.json \
      --region us-west-2
```

3. Recupere a política para seu registro para confirmar.

```
aws ecr get-registry-policy \
      --region us-west-2
```

Conceder permissões de registro para extrair o cache no Amazon ECR

As permissões de registro privado do Amazon ECR podem ser usadas para dimensionar o escopo das permissões de entidades individuais do IAM para usar o cache de pull-through. Se uma entidade do IAM tiver mais permissões concedidas por uma política do IAM do que a política de permissões do registro está concedendo, a política do IAM terá precedência.

Para criar uma política de permissões de um registro privado (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/
2. Na barra de navegação, escolha a região na qual deseja configurar a sua declaração de permissões da política do registro.
3. No painel de navegação, escolha Private registry (Registro privado), Registry permissions (Permissões do registro).
4. Na página Registry permissions (Permissões do registro), escolha Generate statement (Gerar declaração).

5. Para cada declaração de política de permissões de cache de pull-through que você criar, faça o seguinte.

   a. Em Policy type (Tipo de política), escolha Pull through cache policy (Política de cache de pull-through).
   b. Em Statement id (ID da declaração), forneça um nome para a política de declaração do cache de pull-through.
   c. Em IAM entities (Entidades do IAM), especifique os usuários, grupos ou funções a serem incluídos na política.
   d. Em Repository namespace (Namespace do repositório), selecione a regra de cache de pull-through para associar à política.
   e. Em Repository names (Nomes de repositórios), especifique o nome da base do repositório ao qual a regra será aplicada. Por exemplo, se você quisesse especificar o repositório do Amazon Linux no Amazon ECR Public, o nome do repositório seria `amazonlinux`.

Repositórios privados do Amazon ECR

Um repositório privado do Amazon ECR contém suas imagens do Docker, imagens da Open Container Initiative (OCI) e artefatos compatíveis com OCI. Você pode criar, monitorar e excluir repositórios de imagens e definir permissões que controlam quem pode acessá-los usando as operações da API do Amazon ECR ou a seção Repositórios do console do Amazon ECR. O Amazon ECR também se integra à CLI do Docker, para que você possa enviar e extrair imagens de seus ambientes de desenvolvimento para seus repositórios.

Tópicos

Conceitos de repositório privado

- Por padrão, sua conta tem acesso de leitura e gravação aos repositórios no seu registro padrão (*aws_account_id*.dkr.ecr.*region*.amazonaws.com). No entanto, os usuários necessitam de permissões para fazer chamadas para as APIs do Amazon ECR e para enviar e extrair imagens dos repositórios. O Amazon ECR fornece várias políticas gerenciadas para controlar o acesso do usuário em diversos níveis. Para ter mais informações, consulte Exemplos de políticas baseadas em identidade do Amazon Elastic Container Registry.
- Os repositórios podem ser controlados com políticas de acesso de usuários do e políticas de repositório individuais. Para ter mais informações, consulte Políticas de repositório privado no Amazon ECR.
- Os nomes de repositório podem oferecer suporte a namespaces, que você pode usar para agrupar repositórios semelhantes. Por exemplo, se houver diversas equipes usando o mesmo registro, a Equipe A poderia usar o namespace `team-a`, e a Equipe B poderia usar o namespace `team-b`. Ao fazer isso, cada equipe tem sua própria imagem chamada `web-app` com cada imagem prefaciada com o namespace da equipe. Essa configuração permite que essas imagens em cada

equipe sejam usadas simultaneamente sem interferência. A imagem da Equipe A é `team-a/web-app`, e a imagem da Equipe B é `team-b/web-app`.

- Suas imagens podem ser replicadas para outros repositórios nas regiões em seu próprio registro e em todas as contas. Você pode fazer isso especificando uma configuração de replicação nas configurações do Registro. Para ter mais informações, consulte Configurações de registro privado no Amazon ECR.

Criação de um repositório privado do Amazon ECR para armazenar imagens

Crie um repositório privado do Amazon ECR e use o repositório para armazenar suas imagens de contêiner. Siga estas etapas para criar um repositório privado usando o AWS Management Console. Para obter as etapas para criar um repositório usando o AWS CLI, consulteEtapa 3: criar um repositório.

Como criar um repositório (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região na qual criará o seu repositório.
3. Na página Repositórios, escolha Repositórios privados e, em seguida, escolha Criar repositório.
4. Em Visibility settings (Configurações de visibilidade), verifique se Private (Privado) está selecionado.
5. Em Repository name (Nome do repositório), insira um nome exclusivo para o repositório. O nome do repositório pode ser especificado isoladamente (por exemplo, `nginx-web-app`). Como alternativa, pode ter como prefixo um namespace para agrupar o repositório em uma categoria (por exemplo `project-a/nginx-web-app`).

   > **Note**
   >
   > O nome do repositório pode conter até 256 caracteres. O nome de repositório deve começar com uma letra e só pode conter letras minúsculas, números, hifens, sublinhados, pontos e barras. O uso de hífen duplo, sublinhado duplo ou barra dupla não é aceito.

6. Em Tag immutability (Imutabilidade de tag), escolha a configuração de mutabilidade de tag para o repositório. Repositórios configurados com tags imutáveis impedirão que as tags de imagens

sejam sobrescritas. Para ter mais informações, consulte Impedindo que as tags de imagem sejam sobrescritas no Amazon ECR.

7. Em Scan on push (Verificar ao enviar), embora você possa especificar as configurações de verificação no nível do repositório para uma verificação básica, é prática recomendada especificar a configuração de verificação no nível do registro privado. Especificar as configurações de verificação no registro privado permite habilitar a verificação avançada ou a verificação básica, e também definir filtros para especificar quais repositórios são verificados. Para ter mais informações, consulte Digitalize imagens em busca de vulnerabilidades de software no Amazon ECR.
8. Para criptografia KMS, escolha se deseja ativar a criptografia das imagens no repositório usando. AWS Key Management Service Por padrão, quando a criptografia KMS está ativada, o Amazon ECR usa uma Chave gerenciada pela AWS (chave KMS) com o alias. `aws/ecr` Essa chave é criada em sua conta na primeira vez que você cria um repositório com criptografia KMS habilitada. Para ter mais informações, consulte Criptografia em repouso.
9. Quando a criptografia KMS estiver habilitada, selecione Customer encryption settings (advanced) (Configurações de criptografia do cliente (avançado)) para escolher sua própria chave KMS. A chave KMS estar na mesma região que o cluster. Escolha Criar uma AWS KMS chave para navegar até o AWS KMS console e criar sua própria chave.
10. Escolha Criar repositório.

Próximas etapas

Para visualizar as etapas para enviar uma imagem para o seu repositório, selecione o repositório e escolha Exibir comandos push. Para obter mais informações sobre como enviar uma imagem para seu repositório, consulte Enviar uma imagem para um repositório privado do Amazon ECR.

Visualizando o conteúdo e os detalhes de um repositório privado no Amazon ECR

Depois de criar um repositório privado, você pode ver detalhes sobre o repositório no: AWS Management Console

- Quais imagens são armazenadas em um repositório
- Detalhes sobre cada imagem armazenada no repositório, incluindo o tamanho e o resumo SHA para cada imagem

- A frequência de verificação especificada para o conteúdo do repositório
- Se o repositório tem uma regra de cache pull-through ativa associada a ele
- A configuração de criptografia para o repositório

> **Note**
>
> A partir do Docker versão 1.9, o cliente do Docker compacta camadas das imagens antes de enviá-las a um registro do Docker V2. A saída do comando docker images mostra o tamanho da imagem descompactada. Portanto, lembre-se de que o Docker pode retornar uma imagem maior do que a imagem mostrada no AWS Management Console.

Para visualizar as informações do repositório (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região que contém o repositório a ser visualizado.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositories (Repositórios), escolha a guia Private (Privado) e depois o repositório a ser visualizado.
5. Na página de detalhes do repositório, o console usa como padrão a visualização Images (imagens). Use o menu de navegação para visualizar outras informações sobre o repositório.
   - Selecione Summary (Resumo) para visualizar detalhes do repositório e os dados de contagem de extrações do repositório.
   - Escolha Images (Imagens) para visualizar informações sobre etiquetas de imagens no repositório. Para visualizar mais informações sobre a imagem, selecione a etiqueta da imagem. Para ter mais informações, consulte Visualizando detalhes da imagem no Amazon ECR.

     Se houver imagens não marcadas que você deseja excluir, você pode selecionar a caixa à esquerda dos repositórios a serem excluídos e escolha Delete (Excluir). Para ter mais informações, consulte Excluindo uma imagem no Amazon ECR.
   - Escolha Permissões para visualizar as políticas de repositório aplicadas ao repositório. Para ter mais informações, consulte Políticas de repositório privado no Amazon ECR.
   - Escolha Política de ciclo de vida para visualizar as regras de política de ciclo de vida que são aplicadas ao repositório. O histórico de eventos de ciclo de vida também são exibidos

aqui. Para ter mais informações, consulte Automatize a limpeza de imagens usando políticas de ciclo de vida no Amazon ECR.

- Selecione Tags para visualizar as tags de metadados que são aplicadas ao repositório.

Excluindo um repositório privado no Amazon ECR

Se você já terminou de usar um repositório, pode excluí-lo. Quando você exclui um repositório no AWS Management Console, todas as imagens contidas no repositório também são excluídas; isso não pode ser desfeito.

> **Important**
>
> As imagens nos repositórios excluídos também são excluídas. Você não pode desfazer esta operação.

Para excluir um repositório (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região que contém o repositório a ser excluído.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositories (Repositórios), escolha a guia Private (Privado), depois selecione o repositório a excluir e escolha Delete (Excluir).
5. Na janela Excluir ***repository_name***, verifique se os repositórios selecionados devem ser excluídos e escolha Excluir.

Políticas de repositório privado no Amazon ECR

O Amazon ECR usa permissões baseadas em recursos para controlar o acesso a repositórios. As permissões baseadas em recursos permitem que você especifique quais usuários ou funções têm acesso a um repositório e quais ações eles podem realizar no repositório. Por padrão, somente a AWS conta que criou o repositório tem acesso ao repositório. Você pode aplicar uma política de repositório que permita acesso adicional ao seu repositório.

Tópicos

- Políticas de repositório versus políticas do IAM

- Exemplos de políticas de repositório privado no Amazon ECR
- Definindo uma declaração de política de repositório privado no Amazon ECR

Políticas de repositório versus políticas do IAM

As políticas de repositório do Amazon ECR são um subconjunto de políticas do IAM que têm como escopo e são usadas especificamente para controlar o acesso a repositórios individuais do Amazon ECR. As políticas do IAM geralmente são usadas para aplicar permissões a todo o serviço Amazon ECR, mas também podem ser usadas para controlar o acesso a recursos específicos.

Tanto as políticas de repositório do Amazon ECR quanto as políticas do IAM são usadas ao determinar quais ações uma função ou um usuário específico pode executar em um repositório. Se uma função ou um usuário tiver permissão para executar uma ação por meio de uma política de repositório, mas tiver a permissão negada por uma política do IAM (ou vice-versa), a ação será negada. Uma função ou um usuário somente precisa ter permissão para uma ação por meio de uma política de repositório ou uma política do IAM, mas não ambas para que a ação seja permitida.

> **Important**
>
> O Amazon ECR exige que os usuários tenham permissão para fazer chamadas para a API `ecr:GetAuthorizationToken` por meio de uma política do IAM antes que possam fazer a autenticação para um registro e enviar e extrair qualquer imagem de um repositório do Amazon ECR. O Amazon ECR fornece várias políticas gerenciadas do IAM para controlar o acesso do usuário em diversos níveis. Para obter mais informações, consulte Exemplos de políticas baseadas em identidade do Amazon Elastic Container Registry.

Você pode usar qualquer um desses tipos de política para controlar o acesso aos seus repositórios, conforme mostrado nos exemplos a seguir.

Este exemplo mostra uma política de repositório do Amazon ECR que permite que um usuário específico descreva o repositório e as imagens contidas nele.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ECRRepositoryPolicy",
```

```
            "Effect": "Allow",
            "Principal": {"AWS": "arn:aws:iam::account-id:user/username"},
            "Action": [
                "ecr:DescribeImages",
                "ecr:DescribeRepositories"
            ]
        }
    ]
}
```

Este exemplo mostra uma política do IAM que atinge o mesmo objetivo que o acima definindo o escopo da política como um repositório (especificado pelo ARN completo do repositório) usando o parâmetro de recurso. Para obter mais informações sobre o formato do nome de recurso da Amazon (ARN), consulte Recursos.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowDescribeRepoImage",
            "Effect": "Allow",
            "Action": [
                "ecr:DescribeImages",
                "ecr:DescribeRepositories"
            ],
            "Resource": ["arn:aws:ecr:region:account-id:repository/repository-name"]
        }
    ]
}
```

Exemplos de políticas de repositório privado no Amazon ECR

> **Important**
>
> Os exemplos de políticas de repositório nesta página destinam-se a ser aplicados a repositórios privados do Amazon ECR. Eles não funcionarão corretamente se forem usados diretamente com um entidade principal IAM, a menos que sejam modificados para especificar o repositório Amazon ECR como o recurso. Para obter mais informações sobre a definição de políticas de repositório, consulte Definindo uma declaração de política de repositório privado no Amazon ECR.

As políticas de repositório do Amazon ECR são um subconjunto de políticas do IAM que têm como escopo e são usadas especificamente para controlar o acesso a repositórios individuais do Amazon ECR. As políticas do IAM geralmente são usadas para aplicar permissões a todo o serviço Amazon ECR, mas também podem ser usadas para controlar o acesso a recursos específicos. Para ter mais informações, consulte Políticas de repositório versus políticas do IAM.

Os exemplos a seguir de políticas de repositório mostram declarações de permissão que você poderia usar para controlar o acesso aos seus repositórios privados do Amazon ECR.

> **Important**
>
> O Amazon ECR exige que os usuários tenham permissão para fazer chamadas para a API `ecr:GetAuthorizationToken` por meio de uma política do IAM antes que possam fazer a autenticação para um registro e enviar e extrair qualquer imagem de um repositório do Amazon ECR. O Amazon ECR fornece várias políticas gerenciadas do IAM para controlar o acesso do usuário em diversos níveis. Para obter mais informações, consulte Exemplos de políticas baseadas em identidade do Amazon Elastic Container Registry.

Exemplo: permitir um ou mais usuários do

A política de repositório a seguir permite que um ou mais usuários do enviem e extraiam imagens de e para um repositório.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPushPull",
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::account-id:user/push-pull-user-1",
                    "arn:aws:iam::account-id:user/push-pull-user-2"
                ]
            },
            "Action": [
                "ecr:BatchGetImage",
                "ecr:BatchCheckLayerAvailability",
                "ecr:CompleteLayerUpload",
```

```
                "ecr:GetDownloadUrlForLayer",
                "ecr:InitiateLayerUpload",
                "ecr:PutImage",
                "ecr:UploadLayerPart"
            ]
        }
    ]
}
```

Exemplo: permitir outra conta

A política de repositório a seguir permite que uma conta específica insira imagens.

> **Important**
>
> A conta para a qual você está concedendo permissões deve ter a região na qual você está criando a política de repositório ativada, caso contrário, ocorrerá um erro.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCrossAccountPush",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::account-id:root"
            },
            "Action": [
                "ecr:BatchCheckLayerAvailability",
                "ecr:CompleteLayerUpload",
                "ecr:InitiateLayerUpload",
                "ecr:PutImage",
                "ecr:UploadLayerPart"
            ]
        }
    ]
}
```

A política de repositório a seguir permite que alguns usuários extraiam imagens (*pull-user-1* e *pull-user-2*) e forneçam acesso total a outro (*admin-user*).

> **Note**
>
> Para políticas de repositório mais complicadas que atualmente não são suportadas no AWS Management Console, você pode aplicar a política com o set-repository-policy AWS CLI comando.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPull",
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::account-id:user/pull-user-1",
                    "arn:aws:iam::account-id:user/pull-user-2"
                ]
            },
            "Action": [
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer"
            ]
        },
        {
            "Sid": "AllowAll",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::account-id:user/admin-user"
            },
            "Action": [
                "ecr:*"
            ]
        }
    ]
}
```

Exemplo: negar tudo

A política de repositório a seguir nega a todos os usuários a capacidade de extrair imagens.

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "DenyPull",
            "Effect": "Deny",
            "Principal": "*",
            "Action": [
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer"
            ]
        }
    ]
}
```

Exemplo: restringir o acesso a endereços IP específicos

O exemplo a seguir nega permissões a qualquer usuário para executar qualquer operação do Amazon ECR quando aplicada a um repositório de uma faixa específica de endereços.

A condição nesta instrução identifica o intervalo `54.240.143.*` de endereços IP do Internet Protocol versão 4 (IPv4).

O `Condition` bloco usa as `NotIpAddress` condições e a chave de `aws:SourceIp` condição, que é uma chave AWS de condição ampla. Para obter mais informações sobre chaves de condição, consulte Chaves de contexto de condição globais da 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",
    "Id": "ECRPolicyId1",
    "Statement": [
        {
            "Sid": "IPAllow",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "ecr:*",
            "Condition": {
                "NotIpAddress": {
                    "aws:SourceIp": "54.240.143.0/24"
                }
            }
```

```
        }
    ]
}
```

Exemplo: Permitir um AWS serviço

A política de repositório a seguir permite o AWS CodeBuild acesso às ações de API do Amazon ECR necessárias para a integração com esse serviço. Ao usar o exemplo a seguir, você deve usar as chaves de condição `aws:SourceArn` e `aws:SourceAccount` para definir o escopo de quais recursos que podem assumir essas permissões. Para obter mais informações, consulte a amostra do Amazon ECR CodeBuild no Guia do AWS CodeBuild usuário.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"CodeBuildAccess",
         "Effect":"Allow",
         "Principal":{
            "Service":"codebuild.amazonaws.com"
         },
         "Action":[
            "ecr:BatchGetImage",
            "ecr:GetDownloadUrlForLayer"
         ],
         "Condition":{
            "ArnLike":{
               "aws:SourceArn":"arn:aws:codebuild:region:123456789012:project/project-
name"
            },
            "StringEquals":{
               "aws:SourceAccount":"123456789012"
            }
         }
      }
   ]
}
```

Definindo uma declaração de política de repositório privado no Amazon ECR

Você pode adicionar uma declaração de política de acesso a um repositório no AWS Management Console seguindo as etapas abaixo. Você pode adicionar várias instruções de política por repositório. Para obter exemplos de políticas, consulte Exemplos de políticas de repositório privado no Amazon ECR.

> **Important**
>
> O Amazon ECR exige que os usuários tenham permissão para fazer chamadas para a API `ecr:GetAuthorizationToken` por meio de uma política do IAM antes que possam fazer a autenticação para um registro e enviar e extrair qualquer imagem de um repositório do Amazon ECR. O Amazon ECR fornece várias políticas gerenciadas do IAM para controlar o acesso do usuário em diversos níveis. Para obter mais informações, consulte Exemplos de políticas baseadas em identidade do Amazon Elastic Container Registry.

Para configurar uma instrução de política de repositório

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região que contém o repositório no qual será configurada uma instrução de política.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositories (Repositórios), escolha o repositório no qual definir uma instrução de política para visualizar o conteúdo do repositório.
5. Na exibição de lista de imagens do repositório, no painel de navegação, selecione Permissions (Permissões), Edit (Editar).

   > **Note**
   >
   > Se você não vir a opção Permissions (Permissões) no painel de navegação, verifique se você está na exibição de lista de imagens do repositório.

6. Na página Editar permissões, selecione Adicionar declaração.
7. Em Statement name (Nome da instrução), insira um nome para a instrução.

8. Em Effect (Efeito), escolha se a instrução da política resultará em uma permissão ou negação explícita.
9. Em Principal, escolha o escopo ao qual aplicar a instrução da política. Para obter mais informações, consulte AWS Elementos de política JSON: Principal no Manual do usuário do IAM.

   - Você pode aplicar a declaração a todos os AWS usuários autenticados marcando a caixa de seleção Todos (*).
   - Em Service principal (Principal do serviço), especifique o nome do principal do serviço (por exemplo, `ecs.amazonaws.com`) para aplicar a instrução a um serviço específico.
   - Para IDs de AWS conta, especifique um número de AWS conta (por exemplo,`111122223333`) para aplicar a declaração a todos os usuários em uma AWS conta específica. Várias contas podem ser especificadas usando uma lista delimitada por vírgulas.

     Important

     A conta para a qual você está concedendo permissões deve ter a região na qual você está criando a política de repositório ativada, caso contrário, ocorrerá um erro.

   - Para entidades do IAM, selecione as funções ou os usuários em sua AWS conta aos quais aplicar a declaração.

     Note

     Para políticas de repositório mais complicadas que atualmente não são suportadas no AWS Management Console, você pode aplicar a política com o set-repository-policy AWS CLI comando.

10. Em Actions (Ações), escolha o escopo das operações da API do Amazon ECR ao qual a declaração de política deve ser aplicada na lista de operações de API individuais.
11. Quando terminar, escolha Save (Salvar) para definir a política.
12. Repita a etapa anterior para cada política de repositório a ser adicionada.

Marcar um repositório privado no Amazon ECR

Para ajudá-lo a gerenciar seus repositórios Amazon ECR, você pode atribuir seus próprios metadados a repositórios Amazon ECR novos ou existentes usando tags de recursos. AWS Por exemplo, você pode definir um conjunto de tags para os repositórios do Amazon ECR da sua conta para ajudar a rastrear o proprietário de cada repositório.

Conceitos Básicos de Tags

As tags não têm significado semântico no Amazon ECR e são interpretadas estritamente como uma sequência dos caracteres. Tags não são automaticamente atribuídas aos recursos. É possível editar chaves de tags e valores, e é possível remover as tags de um recurso a qualquer momento. É possível definir o valor de uma tag a uma string vazia, mas não pode configurar o valor de um tag como nula. Se você adicionar uma tag que tenha a mesma chave de uma tag existente nesse recurso, o novo valor substituirá o antigo. Se você excluir um recurso, todas as tags do recurso também serão excluídas.

Você pode trabalhar com tags usando o console do Amazon ECR AWS CLI, o e a API do Amazon ECR.

Usando AWS Identity and Access Management (IAM), você pode controlar quais usuários em sua AWS conta têm permissão para criar, editar ou excluir tags. Para obter informações sobre tags nas políticas do IAM, consultethe section called “Usar controle de acesso baseado em tags”.

Marcar recursos para faturamento

As tags que você adiciona aos repositórios do Amazon ECR são úteis para analisar a alocação de custos depois de habilitá-las em seu Relatório de custo e uso. Para ter mais informações, consulte Relatórios de uso do Amazon ECR.

Para ver o custo dos recursos combinados, é possível organizar as informações de faturamento com base nos recursos com os mesmos valores da chave da tag. Por exemplo, é possível etiquetar vários recursos com um nome de aplicação específico, e depois organizar suas informações de faturamento para ver o custo total daquela aplicação em vários serviços. Para obter mais informações sobre como configurar um relatório de alocação de custos com tags, consulte Relatório mensal de alocação de custos no Manual do usuário do AWS Billing .

> **Note**
>
> Se você tiver acabado de habilitar a criação de relatórios, os dados do mês atual estarão disponíveis para visualização após 24 horas.

Adicionar tags a um repositório privado no Amazon ECR

Você pode adicionar tags a um repositório privado.

Para obter informações sobre nomes e práticas recomendadas para tags, consulte Limites e requisitos de nomenclatura de tags e Práticas recomendadas no Guia do usuário de AWS recursos de marcação.

Adicionando tags a um repositório ()AWS Management Console

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/.
2. Na barra de navegação, selecione a região a ser usada.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositórios, marque a caixa de seleção ao lado do repositório que você deseja marcar.
5. No menu Ação, selecione Tags do repositório.
6. Na página Tags do repositório, selecione Adicionar tags, Adicionar tag.
7. Na página Editar tags do repositório, especifique a chave e o valor de cada tag e escolha Salvar.

Adicionar tags a um repositório (AWS CLI ou API)

Você pode adicionar ou substituir uma ou mais tags usando a AWS CLI ou uma API.

- AWS CLI - recurso de tag
- Ação da API - TagResource

Os exemplos a seguir mostram como adicionar tags usando AWS CLI o.

Exemplo 1: Marcar um repositório

O comando a seguir marca um repositório.

```
aws ecr tag-resource \
     --resource-arn arn:aws:ecr:region:account_id:repository/repository_name \
     --tags Key=stack,Value=dev
```

Exemplo 2: Marcar um repositório com várias tags

O comando a seguir adiciona três tags a um repositório.

```
aws ecr tag-resource \
     --resource-arn arn:aws:ecr:region:account_id:repository/repository_name \
     --tags Key=key1,Value=value1 Key=key2,Value=value2 Key=key3,Value=value3
```

Exemplo 3: listar tags de um repositório

O comando a seguir lista as tags associadas a um repositório.

```
aws ecr list-tags-for-resource \
     --resource-arn arn:aws:ecr:region:account_id:repository/repository_name
```

Exemplo 4: criar um repositório e adicionar uma tag

O comando a seguir cria um repositório chamado `test-repo` e adiciona uma tag com a chave `team` e o valor `devs`.

```
aws ecr create-repository \
     --repository-name test-repo \
     --tags Key=team,Value=devs
```

Excluindo tags de um repositório privado no Amazon ECR

Você pode excluir tags de um repositório privado.

Para excluir uma tag de um repositório privado ()AWS Management Console

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/.
2. Na barra de navegação, selecione a região a ser usada.
3. Na página Repositórios, marque a caixa de seleção ao lado do repositório do qual você deseja remover uma tag.
4. No menu Ação, selecione Tags do repositório.

5. Na página Tags do repositório, selecione Editar.
6. Na página Editar tags do repositório, selecione Remover para cada tag que você desejar excluir e escolha Salvar.

Para excluir uma tag de um repositório privado ()AWS CLI

Você pode excluir uma ou mais tags usando a AWS CLI ou uma API.

- AWS CLI - recurso de desmarcação
- Ação da API - UntagResource

O exemplo a seguir mostra como excluir uma tag de um repositório usando o. AWS CLI

```
aws ecr untag-resource \
     --resource-arn arn:aws:ecr:region:account_id:repository/repository_name \
     --tag-keys tag_key
```

Imagens privadas no Amazon ECR

O Amazon ECR armazena imagens do Docker, imagens da Open Container Initiative (OCI) e artefatos compatíveis com OCI em repositórios privados. Você pode usar a CLI do Docker, ou seu cliente preferido, para enviar e extrair imagens dos seus repositórios.

Tópicos

Enviar uma imagem para um repositório privado do Amazon ECR

É possível enviar imagens do Docker, listas de manifesto e imagens da Open Container Initiative (OCI) e artefatos compatíveis para seus repositórios privados.

O Amazon ECR também fornece uma maneira de replicar suas imagens em outros repositórios. Ao especificar uma configuração de replicação nas configurações do seu registro privado, você pode replicar entre regiões em seu próprio registro e em diferentes contas. Para ter mais informações, consulte Configurações de registro privado no Amazon ECR.

Tópicos

- Enviando uma imagem do Docker para um repositório privado do Amazon ECR
- Enviando uma imagem de várias arquiteturas para um repositório privado do Amazon ECR
- Enviando um gráfico do Helm para um repositório privado do Amazon ECR

Permissões do IAM para enviar uma imagem para um repositório privado do Amazon ECR

Os usuários precisam de permissões do IAM para enviar imagens para os repositórios privados do Amazon ECR. Seguindo a melhor prática de conceder privilégios mínimos, você pode conceder acesso a um repositório específico. Você também pode conceder acesso a todos os repositórios.

Um usuário deve se autenticar em cada registro do Amazon ECR para o qual deseja enviar imagens solicitando um token de autorização. O Amazon ECR fornece várias políticas AWS gerenciadas para controlar o acesso do usuário em vários níveis. Para ter mais informações, consulte AWS políticas gerenciadas para o Amazon Elastic Container Registry.

Você também pode criar suas próprias políticas de IAM. A política do IAM a seguir concede as permissões necessárias para enviar uma imagem para um repositório específico. O repositório deve ser especificado como um nome do recurso da Amazon (ARN) completo.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ecr:CompleteLayerUpload",
                "ecr:UploadLayerPart",
                "ecr:InitiateLayerUpload",
                "ecr:BatchCheckLayerAvailability",
                "ecr:PutImage"
            ],
            "Resource": "arn:aws:ecr:region:111122223333:repository/repository-name"
        },
        {
            "Effect": "Allow",
            "Action": "ecr:GetAuthorizationToken",
            "Resource": "*"
        }
    ]
```

```
}
```

A política do IAM a seguir concede as permissões necessárias para enviar uma imagem para todos os repositórios.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ecr:CompleteLayerUpload",
                "ecr:GetAuthorizationToken",
                "ecr:UploadLayerPart",
                "ecr:InitiateLayerUpload",
                "ecr:BatchCheckLayerAvailability",
                "ecr:PutImage"
            ],
            "Resource": "*"
        }
    ]
}
```

Enviando uma imagem do Docker para um repositório privado do Amazon ECR

Você pode enviar suas imagens de contêiner para um repositório do Amazon ECR com o comando docker push.

O Amazon ECR também oferece suporte à criação e envio de listas de manifestos do Docker que são usadas para imagens de várias arquiteturas. Para mais informações, consulte Enviando uma imagem de várias arquiteturas para um repositório privado do Amazon ECR.

Para enviar uma imagem do Docker a um repositório do Amazon ECR

O repositório do Amazon ECR deve existir antes de enviar a imagem. Para ter mais informações, consulte the section called “Criação de um repositório para armazenar imagens”.

1. Autentique o cliente do Docker para o registro do Amazon ECR para o qual você pretende enviar a imagem. Os tokens de autenticação devem ser obtidos para cada registro usado e são válidos

por 12 horas. Para ter mais informações, consulte Autenticação de registro privado no Amazon ECR.

Para autenticar o Docker para um registro do Amazon ECR, execute o comando aws ecr get-login-password. Ao transmitir o token de autenticação para o comando docker login, use o valor `AWS` para o nome de usuário, e especifique o URI de registro do Amazon ECR para o qual deseja fazer a autenticação. Se autenticar em vários registros, você deverá repetir o comando para cada registro.

> **Important**
>
> Se você receber um erro, instale ou atualize para a versão mais recente da AWS CLI. Para obter mais informações, consulte Instalar a AWS Command Line Interface no Guia do usuário da AWS Command Line Interface .

```
aws ecr get-login-password --region region | docker login --username AWS --password-stdin aws_account_id.dkr.ecr.region.amazonaws.com
```

2. Se o seu repositório de imagens não existir no registro que você pretende enviar, crie-o. Para ter mais informações, consulte Criação de um repositório privado do Amazon ECR para armazenar imagens.
3. Identifique a imagem a ser enviada. Execute o comando docker images para listar as imagens do contêiner em seu sistema.

```
docker images
```

Você pode identificar uma imagem com o valor *repository:tag* ou o ID da imagem na saída de comando resultante.

4. Marque a sua imagem com o registro do Amazon ECR, o repositório e a combinação opcional de nomes de tag de imagem a ser usada. O formato do registro é *aws_account_id*.dkr.ecr.*us-west-2*.amazonaws.com. O nome do repositório deve corresponder ao repositório que você criou para sua imagem. Se você omitir a tag de imagem, suporemos que a tag é `latest`.

   O exemplo a seguir marca uma imagem local com o ID *e9ae3c220b23* como *aws_account_id*.dkr.ecr.*us-west-2*.amazonaws.com/my-repository:tag.

```
docker tag e9ae3c220b23 aws_account_id.dkr.ecr.us-west-2.amazonaws.com/my-repository:tag
```

5. Envie a imagem usando o comando docker push:

```
docker push aws_account_id.dkr.ecr.us-west-2.amazonaws.com/my-repository:tag
```

6. (Opcional) Aplique quaisquer tags adicionais à sua imagem e envie essas tags ao Amazon ECR repetindo Step 4 e Step 5.

Enviando uma imagem de várias arquiteturas para um repositório privado do Amazon ECR

Você pode enviar imagens de várias arquiteturas para um repositório Amazon ECR criando e enviando listas de manifestos do Docker. Uma lista de manifestos é uma lista de imagens criada com a especificação de um ou mais nomes de imagem. Na maioria dos casos, a lista de manifestos é criada a partir de imagens que têm a mesma função, mas são para sistemas operacionais ou arquiteturas diferentes. A lista de manifestos não é obrigatória. Para obter mais informações, consulte manifesto do docker.

Uma lista de manifestos pode ser extraída ou referenciada em uma definição de tarefa do Amazon ECS ou especificação de pod do Amazon EKS como outras imagens do Amazon ECR.

Pré-requisitos

- Na CLI do Docker, ative os recursos experimentais. Para obter informações sobre recursos experimentais, consulte Recursos experimentais na documentação do Docker.
- O repositório do Amazon ECR deve existir antes de enviar a imagem. Para ter mais informações, consulte the section called “Criação de um repositório para armazenar imagens”.
- As imagens devem ser enviadas ao seu repositório antes de você criar o manifesto do Docker. Para obter informações sobre como enviar uma imagem, consulte Enviando uma imagem do Docker para um repositório privado do Amazon ECR.

Como enviar uma imagem de multiarquitetura do Docker para um repositório do Amazon ECR

1. Autentique o cliente do Docker para o registro do Amazon ECR para o qual você pretende enviar a imagem. Os tokens de autenticação devem ser obtidos para cada registro usado e são válidos

por 12 horas. Para ter mais informações, consulte Autenticação de registro privado no Amazon ECR.

Para autenticar o Docker para um registro do Amazon ECR, execute o comando aws ecr get-login-password. Ao transmitir o token de autenticação para o comando docker login, use o valor `AWS` para o nome de usuário, e especifique o URI de registro do Amazon ECR para o qual deseja fazer a autenticação. Se autenticar em vários registros, você deverá repetir o comando para cada registro.

> **Important**
>
> Se você receber um erro, instale ou atualize para a versão mais recente da AWS CLI. Para obter mais informações, consulte Installing the AWS Command Line Interface (Instalar a AWS Command Line Interface) no User Guide (Guia do usuário da).

```
aws ecr get-login-password --region region | docker login --username AWS --
password-stdin aws_account_id.dkr.ecr.region.amazonaws.com
```

2. Listar as imagens no repositório, confirmando as tags de imagem.

```
aws ecr describe-images --repository-name my-repository
```

3. Criar a lista de manifestos do Docker. O comando `manifest create` verifica se as imagens referenciadas já estão no repositório e cria o manifesto localmente.

```
docker manifest create aws_account_id.dkr.ecr.us-west-2.amazonaws.com/my-
repository aws_account_id.dkr.ecr.us-west-2.amazonaws.com/my-
repository:image_one_tag aws_account_id.dkr.ecr.us-west-2.amazonaws.com/my-
repository:image_two
```

4. (Opcional) Inspecionar a lista de manifestos do Docker. Isso permite que você confirme o tamanho e o resumo de cada manifesto de imagem referenciado na lista de manifestos.

```
docker manifest inspect aws_account_id.dkr.ecr.us-west-2.amazonaws.com/my-
repository
```

5. Enviar a lista de manifestos do Docker para seu repositório do Amazon ECR.

```
docker manifest push aws_account_id.dkr.ecr.us-west-2.amazonaws.com/my-repository
```

Enviando um gráfico do Helm para um repositório privado do Amazon ECR

Você pode enviar artefatos da Open Container Initiative (OCI) para um repositório Amazon ECR. Para ver um exemplo dessa funcionalidade, use as etapas a seguir para enviar um gráfico do Helm para o Amazon ECR.

Para obter informações sobre como usar seus gráficos Helm hospedados no Amazon ECR com o Amazon EKS, consulte. Instalação de um gráfico do Helm em um cluster Amazon EKS

Para enviar um chart do Helm para um repositório do Amazon ECR

1. Use a versão mais recente do cliente do Helm. Estas etapas foram escritas usando a versão `3.8.2` do Helm. Para obter mais informações, consulte Instalação do Helm.
2. Use as etapas a seguir para criar um chart do Helm. Para obter mais informações, consulte o Documentos do Helm - Introdução.

   a. Crie um chart do Helm denominado `helm-test-chart` e limpe o conteúdo da caixa do diretório `templates`.

   ```
   helm create helm-test-chart
   rm -rf ./helm-test-chart/templates/*
   ```

   b. Crie um ConfigMap na `templates` pasta.

   ```
   cd helm-test-chart/templates
   cat <<EOF > configmap.yaml
   apiVersion: v1
   kind: ConfigMap
   metadata:
     name: helm-test-chart-configmap
   data:
     myvalue: "Hello World"
   EOF
   ```

3. Embalar o gráfico. A saída conterá o nome do arquivo do chart empacotado que você usa ao enviar o chart do Helm.

```
cd ../..
helm package helm-test-chart
```

Saída

```
Successfully packaged chart and saved it to: /Users/username/helm-test-
chart-0.1.0.tgz
```

4. Crie um repositório para armazenar o chart do Helm. O nome do repositório deve corresponder ao nome utilizado ao criar o chart do Helm na etapa 2. Para ter mais informações, consulte Criação de um repositório privado do Amazon ECR para armazenar imagens.

```
aws ecr create-repository \
     --repository-name helm-test-chart \
     --region us-west-2
```

5. Autentique o cliente do Helm para o registro do Amazon ECR para o qual você pretende enviar o chart do Helm. Os tokens de autenticação devem ser obtidos para cada registro usado e são válidos por 12 horas. Para ter mais informações, consulte Autenticação de registro privado no Amazon ECR.

```
aws ecr get-login-password \
     --region us-west-2 | helm registry login \
     --username AWS \
     --password-stdin aws_account_id.dkr.ecr.us-west-2.amazonaws.com
```

6. Envie o chart do Helm usando o comando helm push. A saída deve incluir o URI do repositório do Amazon ECR e o resumo do SHA.

```
helm push helm-test-chart-0.1.0.tgz oci://aws_account_id.dkr.ecr.us-
west-2.amazonaws.com/
```

7. Descreva seu chart do Helm.

```
aws ecr describe-images \
     --repository-name helm-test-chart \
     --region us-west-2
```

Na saída, verifique se o parâmetro `artifactMediaType` indica o tipo de artefato apropriado.

```
{
    "imageDetails": [
        {
            "registryId": "aws_account_id",
            "repositoryName": "helm-test-chart",
            "imageDigest":
 "sha256:dd8aebdda7df991a0ffe0b3d6c0cf315fd582cd26f9755a347a52adEXAMPLE",
            "imageTags": [
                "0.1.0"
            ],
            "imageSizeInBytes": 1620,
            "imagePushedAt": "2021-09-23T11:39:30-05:00",
            "imageManifestMediaType": "application/vnd.oci.image.manifest.v1+json",
            "artifactMediaType": "application/vnd.cncf.helm.config.v1+json"
        }
    ]
}
```

8. (Opcional) Para etapas adicionais, instale o configmap do Helm e comece a usar o Amazon EKS. Para ter mais informações, consulte Instalação de um gráfico do Helm em um cluster Amazon EKS.

Assinatura de uma imagem armazenada em um repositório privado do Amazon ECR

O Amazon ECR se integra AWS Signer para fornecer uma maneira de você assinar suas imagens de contêiner. Você pode armazenar as imagens do contêiner e as assinaturas em seus repositórios privados.

Considerações

As seguintes informações devem ser consideradas ao usar a assinatura de imagens do Amazon ECR.

- As assinaturas armazenadas em seu repositório contam para a cota do serviço do número máximo de imagens por repositório. Para ter mais informações, consulte Cotas de serviço do Amazon ECR.

- Ao usar as políticas de ciclo de vida do Amazon ECR, qualquer ação de uma regra para expirar ou excluir um índice de imagem OCI fará com que o Amazon ECR exclua todas as assinaturas com referência a esse índice de imagem em 24 horas.

Pré-requisitos

Antes de começar, certifique-se de que os seguintes pré-requisitos sejam atendidos.

- Instale e configure a versão mais recente do AWS CLI. Para obter mais informações, consulte Instalar ou atualizar a versão mais recente da AWS CLI no Guia do usuário do AWS Command Line Interface .
- Instale a CLI do Notation e o plug-in para AWS Signer o Notation. Para obter mais informações, consulte Pré-requisitos para assinar imagens de contêiner no Guia do desenvolvedor do AWS Signer .
- Tenha uma imagem de contêiner armazenada em um repositório privado do Amazon ECR para assinar. Para ter mais informações, consulte Enviar uma imagem para um repositório privado do Amazon ECR.

Configurar autenticação para o cliente do Notary

Antes de criar uma assinatura usando a CLI do Notation, você deve configurar o cliente para que ele possa realizar autenticação no Amazon ECR. Se o Docker estiver instalado no mesmo host em que você instalou o cliente do Notation, o Notation reutilizará o mesmo método de autenticação usado para o cliente do Docker. Os comandos `login` e `logout` permitirão que os comandos `sign` e `verify` do Notation usem essas mesmas credenciais, e você não precisará autenticar o Notation separadamente. Para obter mais informações sobre como configurar o cliente do Notation para autenticação, consulte Authenticate with OCI-compliant registries na documentação do projeto Notary

Se você não estiver usando o Docker ou outra ferramenta que use as credenciais do Docker, recomendamos que use o auxiliar de credenciais do Docker doo Amazon ECR como repositório de credenciais. Para obter mais informações sobre como instalar e configurar o auxiliar de credenciais do Amazon ECR, consulte Amazon ECR Docker Credential Helper.

Assinar uma imagem

As etapas a seguir podem ser usadas para criar os recursos necessários para assinar uma imagem de contêiner e armazenar a assinatura em um repositório privado do Amazon ECR. O Notation assina imagens usando o resumo.

Para assinar uma imagem

1. Crie um perfil de AWS Signer assinatura usando a plataforma de `Notation-OCI-SHA384-ECDSA` assinatura. Você também pode especificar um período de validade de assinatura usando o parâmetro `--signature-validity-period`. Esse valor pode ser especificado usando `DAYS`, `MONTHS` ou `YEARS`. Se nenhum período de validade for especificado, será usado o valor padrão de 135 meses.

   ```
   aws signer put-signing-profile --profile-name ecr_signing_profile --platform-id
    Notation-OCI-SHA384-ECDSA
   ```

   > **Note**
   >
   > O nome do perfil de assinatura só aceita caracteres alfanuméricos e sublinhado (_).

2. Autentique o cliente do Notation em seu registro padrão. O exemplo a seguir usa o AWS CLI para autenticar o Notation CLI em um registro privado do Amazon ECR.

   ```
   aws ecr get-login-password --region region | notation login --username AWS --
   password-stdin 111122223333.dkr.ecr.region.amazonaws.com
   ```

3. Use o Notation CLI para assinar a imagem, especificando a imagem usando o nome do repositório e o resumo SHA. Isso cria a assinatura e a envia para o mesmo repositório privado do Amazon ECR onde está a imagem que está sendo assinada.

   No exemplo a seguir, assinaremos uma imagem no repositório `curl` usando o resumo SHA `sha256:ca78e5f730f9a789ef8c63bb55275ac12dfb9e8099e6EXAMPLE`.

   ```
   notation
    sign 111122223333.dkr.ecr.region.amazonaws.com/
   curl@sha256:ca78e5f730f9a789ef8c63bb55275ac12dfb9e8099e6EXAMPLE --plugin
    "com.amazonaws.signer.notation.plugin" --id "arn:aws:signer:region:111122223333:/
   signing-profiles/ecrSigningProfileName"
   ```

Próximas etapas

Depois de assinar a imagem do contêiner, você pode verificar a assinatura localmente. Para obter instruções sobre como verificar uma imagem, consulte Verificar uma imagem localmente após fazer login no Guia do AWS Signer desenvolvedor

Excluindo uma assinatura de um repositório privado do Amazon ECR

Você pode excluir uma assinatura de um repositório privado do Amazon ECR. Quando você cria e envia uma assinatura usando o Notation CLI, um índice de imagem OCI também é criado em seu repositório do Amazon ECR. A API do Amazon ECR não aceita a exclusão de artefatos ou imagens com referência de um índice de imagem da OCI, portanto, a seguir estão as opções disponíveis para limpar esses artefatos.

- (Recomendado) Você pode usar a CLI do ORAS para excluir o artefato e o ORAS cuidará da atualização ou exclusão do índice da imagem.
- Você pode usar a API ou o console do Amazon ECR para excluir primeiro o índice de imagem da OCI e depois o artefato referenciado, como a assinatura.

Ao usar o cliente do ORAS para excluir assinaturas e outros artefatos do tipo de referência, o ORAS gerencia o índice de imagens OCI. O ORAS primeiro removerá a referência ao artefato do índice e, em seguida, excluirá o manifesto. O comando `oras manifest delete` pode ser usado fazendo referência ao índice do artefato de assinatura.

Para excluir uma assinatura usando a CLI do ORAS

1. Instale e configure o cliente ORAS.

   Para obter informações sobre como instalar e configurar o cliente ORAS, consulte Instalação na documentação do ORAS.

2. Para excluir uma assinatura usando a CLI do ORAS, execute o seguinte comando:

```
oras manifest
 delete 111122223333.dkr.ecr.region.amazonaws.com/
repository_name@sha256:ca78e5f730f9a789ef8c63bb55275ac12dfb9e8099e6EXAMPLE
```

Visualizando detalhes da imagem no Amazon ECR

Depois de enviar uma imagem para o seu repositório, você pode ver as informações sobre ela. Os detalhes incluídos são os seguintes:

- URI da imagem
- Tags da imagem
- Tipo de mídia do Artifact
- Tipo de manifesto da imagem
- Status da verificação
- O tamanho da imagem em MB
- Quando a imagem foi extraída para o repositório
- O status da replicação

Para ver os detalhes da imagem (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região que contém o repositório de sua imagem.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositórios, escolha o repositório a ser visualizado.
5. Na página Repositórios: ***`nome do repositório`***, escolha a imagem cujos detalhes você deseja visualizar.

Extrair uma imagem para seu ambiente local a partir de um repositório privado do Amazon ECR

Se quiser executar uma imagem do Docker que está disponível no Amazon ECR, você pode extrai-la para seu ambiente local com o comando docker pull. Você pode fazer isso a partir do seu registro padrão ou de um registro associado a outra AWS conta.

Para usar uma imagem do Amazon ECR em uma definição de tarefa do Amazon ECS, consulte Uso de imagens do Amazon ECR com o Amazon ECS.

> **Important**
>
> O Amazon ECR exige que os usuários tenham permissão para fazer chamadas para a API `ecr:GetAuthorizationToken` por meio de uma política do IAM antes que possam fazer a autenticação para um registro e enviar e extrair qualquer imagem de um repositório do Amazon ECR. O Amazon ECR fornece várias políticas AWS gerenciadas para controlar o acesso do usuário em vários níveis. Para obter informações sobre as políticas AWS gerenciadas do Amazon ECR, consulteAWS políticas gerenciadas para o Amazon Elastic Container Registry.

Para extrair uma imagem do Docker de um repositório do Amazon ECR

1. Autentique o cliente do Docker para o registro do Amazon ECR do qual você pretende extrair a imagem. Os tokens de autenticação devem ser obtidos para cada registro usado e são válidos por 12 horas. Para ter mais informações, consulte Autenticação de registro privado no Amazon ECR.
2. (Opcional) Identifique a imagem a ser extraída.
   - É possível listar os repositórios em um registro com o comando aws ecr describe-repositories:

     ```
     aws ecr describe-repositories
     ```

     O registro de exemplo acima tem um repositório chamado `amazonlinux`.
   - É possível descrever as imagens em um repositório com o comando aws ecr describe-images:

     ```
     aws ecr describe-images --repository-name amazonlinux
     ```

     O repositório de exemplo acima tem uma imagem marcada como `latest` e `2016.09`, com o resumo de imagem `sha256:f1d4ae3f7261a72e98c6ebefe9985cf10a0ea5bd762585a43e0700ed99863807`.
3. Extraia a imagem usando o comando docker pull. O formato de nome de imagem deve ser *registry*/*repository*[:*tag*] para extrair por tag ou *registry*/*repository*[@*digest*] para extrair por resumo.

```
docker pull aws_account_id.dkr.ecr.us-west-2.amazonaws.com/amazonlinux:latest
```

> **Important**
>
> Se receber um erro `repository-url not found: does not exist or no pull access`, pode ser necessário autenticar seu cliente do Docker com o Amazon ECR. Para ter mais informações, consulte Autenticação de registro privado no Amazon ECR.

Extraindo a imagem do contêiner Amazon Linux

A imagem do contêiner do Amazon Linux é criada a partir dos mesmos componentes de software que são incluídos na AMI do Amazon Linux. A imagem do contêiner Amazon Linux está disponível para uso em qualquer ambiente como imagem base para cargas de trabalho do Docker. Se você usa o Amazon Linux AMI para aplicativos no Amazon EC2, você pode conteinerizar seus aplicativos com a imagem de contêiner Amazon Linux.

Você pode usar a imagem do contêiner Amazon Linux em seu ambiente de desenvolvimento local e, em seguida, enviar seu aplicativo para AWS usar o Amazon ECS. Para ter mais informações, consulte Uso de imagens do Amazon ECR com o Amazon ECS.

A imagem de contêiner do Amazon Linux está disponível no Amazon ECR Public no Docker Hub. Para obter suporte para a imagem do contêiner Amazon Linux, acesse os fóruns de AWS desenvolvedores.

Para extrair a imagem de contêiner do Amazon Linux do Amazon ECR Public

1. Autentique o cliente do Docker para seu registro do Amazon Linux. Os tokens de autenticação são válidos por 12 horas. Para ter mais informações, consulte Autenticação de registro privado no Amazon ECR.

   > **Note**
   >
   > Os comandos ecr-public estão disponíveis na AWS CLI a partir da versão `1.18.1.187`. No entanto, recomendamos usar a versão mais recente da AWS CLI. Para obter mais informações, consulte Instalar a AWS Command Line Interface no Guia do usuário da AWS Command Line Interface .

```
aws ecr-public get-login-password --region us-east-1 | docker login --username AWS
 --password-stdin public.ecr.aws
```

A saída é a seguinte:

```
Login succeeded
```

2. Extraia a imagem do contêiner do Amazon Linux usando o comando docker pull. Para visualizar a imagem do contêiner do Amazon Linux na Galeria Pública do Amazon ECR, consulte Galeria pública do Amazon ECR - amazonlinux.

```
docker pull public.ecr.aws/amazonlinux/amazonlinux:latest
```

3. (Opcional) Execute o contêiner localmente.

```
docker run -it public.ecr.aws/amazonlinux/amazonlinux /bin/bash
```

Para extrair a imagem do contêiner do Amazon Linux a partir do Docker Hub

1. Extraia a imagem do contêiner do Amazon Linux usando o comando docker pull.

```
docker pull amazonlinux
```

2. (Opcional) Execute o contêiner localmente.

```
docker run -it amazonlinux:latest /bin/bash
```

Excluindo uma imagem no Amazon ECR

Se você já terminou de usar uma imagem, pode excluí-la do repositório. Se você já terminou de usar um repositório, pode excluir o repositório inteiro e todas as imagens contidas nele. Para ter mais informações, consulte Excluindo um repositório privado no Amazon ECR.

Como alternativa a excluir as imagens manualmente, você pode criar políticas de ciclo de vida do repositório que fornecem mais controle sobre o gerenciamento do ciclo de vida das imagens em seus repositórios. As políticas de ciclo de vida automatizam esse processo para você. Para ter mais

informações, consulte Automatize a limpeza de imagens usando políticas de ciclo de vida no Amazon ECR.

Para excluir uma imagem (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região que contém a imagem a ser excluída.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositórios, escolha o repositório que contém a imagem a ser excluída.
5. Na página Repositórios: ***repository_name***, selecione a caixa à esquerda da imagem a ser excluídas e escolha Excluir.
6. Na caixa de diálogo Excluir imagem(ns), verifique se as imagens selecionadas devem ser excluídas e escolha Excluir.

Para excluir uma imagem (AWS CLI)

1. Listar as imagens no seu repositório. As imagens marcadas terão um resumo de imagem, bem como uma lista de tags associadas. Imagens não marcadas só terão um resumo de imagem.

```
aws ecr list-images \
     --repository-name my-repo
```

2. (Opcional) Exclua quaisquer tags indesejáveis da imagem especificando a tag da imagem que você deseja excluir. Quando você excluir a última tag de uma imagem, a imagem será excluída.

```
aws ecr batch-delete-image \
     --repository-name my-repo \
     --image-ids imageTag=tag1 imageTag=tag2
```

3. Exclua uma imagem marcada ou não marcada especificando o resumo da imagem. Quando você excluir uma imagem fazendo referência ao seu resumo, a imagem e todas as suas tags serão excluídas.

```
aws ecr batch-delete-image \
     --repository-name my-repo \
     --image-ids imageDigest=sha256:4f70ef7a4d29e8c0c302b13e25962d8f7a0bd304EXAMPLE
```

Para excluir várias imagens, você pode especificar várias tags de imagem ou resumos de imagem na solicitação.

```
aws ecr batch-delete-image \
     --repository-name my-repo \
     --image-ids imageDigest=sha256:4f70ef7a4d29e8c0c302b13e25962d8f7a0bd304EXAMPLE
 imageDigest=sha256:f5t0e245ssffc302b13e25962d8f7a0bd304EXAMPLE
```

Como remarcar uma imagem no Amazon ECR

Com as imagens do esquema 2 do manifesto de imagem do Docker V2, você pode usar a opção `--image-tag` do comando put-image para remarcar uma imagem existente. Você pode remarcar sem extrair ou enviar a imagem com Docker. Para imagens maiores, esse processo economiza uma quantidade considerável de largura de banda e de tempo necessário para remarcar uma imagem.

Como remarcar uma imagem (AWS CLI)

Para remarcar uma imagem com a AWS CLI

1. Use o comando batch-get-image para obter o manifesto da imagem para remarcá-la e gravá-la em um arquivo. Neste exemplo, o manifesto de uma imagem com a tag *latest* no repositório *amazonlinux* é gravado em uma variável de ambiente chamada *MANIFEST*.

   ```
   MANIFEST=$(aws ecr batch-get-image --repository-name amazonlinux --image-ids
    imageTag=latest --output text --query 'images[].imageManifest')
   ```

2. Use a opção `--image-tag` do comando put-image para colocar o manifesto da imagem no Amazon ECR com uma nova tag. Neste exemplo, a imagem é marcada como *2017.03*.

   > **Note**
   >
   > Se a `--image-tag` opção não estiver disponível na sua versão do AWS CLI, atualize para a versão mais recente. Para obter mais informações, consulte Instalar a AWS Command Line Interface no Guia do usuário da AWS Command Line Interface .

```
aws ecr put-image --repository-name amazonlinux --image-tag 2017.03 --image-
manifest "$MANIFEST"
```

3. Verifique se a sua nova tag de imagem está conectada à imagem. Na saída a seguir, a imagem têm as tags `latest` e `2017.03`.

```
aws ecr describe-images --repository-name amazonlinux
```

A saída é a seguinte:

```
{
    "imageDetails": [
        {
            "imageSizeInBytes": 98755613,
            "imageDigest":
 "sha256:8d00af8f076eb15a33019c2a3e7f1f655375681c4e5be157a26EXAMPLE",
            "imageTags": [
                "latest",
                "2017.03"
            ],
            "registryId": "aws_account_id",
            "repositoryName": "amazonlinux",
            "imagePushedAt": 1499287667.0
        }
    ]
}
```

Como remarcar uma imagem (AWS Tools for Windows PowerShell)

Para remarcar uma imagem com a AWS Tools for Windows PowerShell

1. Use o cmdlet Get-ECRImageBatch para obter a descrição da imagem para remarcá-la e gravá-la em uma variável de ambiente. Neste exemplo, uma imagem com a tag, *`latest`*, no repositório, *`amazonlinux`*, é gravada na variável de ambiente, *`$Image`*.

> **Note**
>
> Se você não tiver o cmdlet Get-ECRImageBatch disponível no sistema, consulte Configuração do AWS Tools for Windows PowerShell no Manual do usuário do AWS Tools for Windows PowerShell .

```
$Image = Get-ECRImageBatch -ImageId @{ imageTag="latest" } -
RepositoryName amazonlinux
```

2. Escreva o manifesto de imagem para a variável de ambiente *$Manifest*.

```
$Manifest = $Image.Images[0].ImageManifest
```

3. Use a opção `-ImageTag` do cmdlet Write-ECRImage para colocar o manifesto da imagem no Amazon ECR com uma nova tag. Neste exemplo, a imagem é marcada como *2017.09*.

```
Write-ECRImage -RepositoryName amazonlinux -ImageManifest $Manifest -
ImageTag 2017.09
```

4. Verifique se a sua nova tag de imagem está conectada à imagem. Na saída a seguir, a imagem têm as tags `latest` e `2017.09`.

```
Get-ECRImage -RepositoryName amazonlinux
```

A saída é a seguinte:

```
ImageDigest                                                             ImageTag
-----------                                                             --------
sha256:359b948ea8866817e94765822787cd482279eed0c17bc674a7707f4256d5d497 latest
sha256:359b948ea8866817e94765822787cd482279eed0c17bc674a7707f4256d5d497 2017.09
```

Impedindo que as tags de imagem sejam sobrescritas no Amazon ECR

Você pode evitar que as tags de imagem sejam sobrescritas ativando a imutabilidade da tag em um repositório. Depois que a imutabilidade da tag é ativada, o `ImageTagAlreadyExistsException` erro é retornado se você enviar uma imagem com uma tag que já está no repositório. A imutabilidade da tag afeta todas as tags. Você não pode tornar algumas tags imutáveis e outras não.

Você pode usar as AWS CLI ferramentas AWS Management Console e para definir a mutabilidade da tag de imagem para um novo repositório ou para um repositório existente. Para criar um repositório usando as etapas do console, consulteCriação de um repositório privado do Amazon ECR para armazenar imagens.

Configurando a mutabilidade da tag de imagem ()AWS Management Console

Para definir a mutabilidade da tag de imagem

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região que contém o repositório a ser editado.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositórios, escolha a guia Privado, depois selecione o repositório a editar e escolha Editar.
5. Em Tag immutability (Imutabilidade de tag), escolha a configuração de mutabilidade de tag para o repositório. Repositórios configurados com tags imutáveis impedirão que as tags de imagens sejam sobrescritas. Para ter mais informações, consulte Impedindo que as tags de imagem sejam sobrescritas no Amazon ECR.
6. Em Image scan settings (Configurações de verificação de imagens), embora você possa especificar as configurações de verificação no nível do repositório para uma verificação básica, é prática recomendada especificar a configuração de verificação no nível do registro privado. Especificar as configurações de verificação no registro privado permite habilitar a verificação avançada ou a verificação básica, e também definir filtros para especificar quais repositórios são verificados. Para ter mais informações, consulte Digitalize imagens em busca de vulnerabilidades de software no Amazon ECR.

7. Em Encryption settings (Configurações de criptografia), este é um campo somente para visualização, pois as configurações de criptografia de um repositório não podem ser alteradas depois que ele é criado.
8. Escolha Save (Salvar) para atualizar as configurações do repositório.

Configurando a mutabilidade da tag de imagem ()AWS CLI

Como criar um repositório com tags imutáveis configuradas

Use um dos comandos a seguir para criar um novo repositório de imagens com tags imutáveis configuradas.

- create-repository (AWS CLI)

```
aws ecr create-repository --repository-name name --image-tag-mutability IMMUTABLE --
region us-east-2
```

- New-ECRRepository (AWS Tools for Windows PowerShell)

```
New-ECRRepository -RepositoryName name -ImageTagMutability IMMUTABLE -Region us-
east-2 -Force
```

Para atualizar as configurações de mutabilidade da tag de imagem para um repositório

Use um dos comandos a seguir para atualizar as configurações de mutabilidade de tag de imagem de um repositório existente.

- put-image-tag-mutability (AWS CLI)

```
aws ecr put-image-tag-mutability --repository-name name --image-tag-
mutability IMMUTABLE --region us-east-2
```

- Mutabilidade Write-ECR () ImageTagAWS Tools for Windows PowerShell

```
Write-ECRImageTagMutability -RepositoryName name -ImageTagMutability IMMUTABLE -
Region us-east-2 -Force
```

Suporte ao formato de manifesto de imagem de contêiner no Amazon ECR

O Amazon ECR oferece suporte aos seguintes formatos de manifesto de imagem de contêiner:

- Schema 1 de manifesto V2 de imagem de Docker (usado com o Docker versão 1.9 e anteriores)
- Schema 2 de manifesto V2 de imagem de Docker (usado com o Docker versão 1.10 e posteriores)
- Especificações de Open Container Initiative (OCI – Iniciativa de contêiner aberto) (v1.0 e posteriores)

O suporte para o schema 2 de manifesto V2 de imagem de Docker fornece a seguinte funcionalidade:

- A possibilidade de usar várias tags para uma única imagem.
- Suporte para armazenar imagens de contêiner do Windows.

Conversão de manifesto de imagem do Amazon ECR

Quando você envia imagens para o Amazon ECR e extrai imagens do Amazon ECR, o cliente de mecanismo de contêiner (por exemplo, Docker) se comunica com o registro para acordar um formato de manifesto que seja entendido pelo cliente e pelo registro para ser usado na imagem.

Quando você envia uma imagem ao Amazon ECR com o Docker versão 1.9 ou anterior, o formato do manifesto da imagem é armazenado como manifesto de imagem do Docker V2 esquema 1. Quando você envia uma imagem ao Amazon ECR com o Docker versão 1.10 ou posterior, o formato do manifesto da imagem é armazenado como manifesto de imagem do Docker V2 esquema 2.

Quando você extrai uma imagem do Amazon ECR por tag, o Amazon ECR retorna o formato de manifesto de imagem que está armazenado no repositório. Mas somente se o formato é entendido pelo cliente. Se o formato do manifesto de imagem armazenado não é entendido pelo cliente, o Amazon ECR converte o manifesto de imagem em um formato que seja entendido pelo cliente. Por exemplo, se um cliente do Docker 1.9 solicitar um manifesto de imagem armazenado como manifesto de imagem do Docker V2 esquema 2, o Amazon ECR devolve-o no formato de manifesto de imagem do Docker V2 esquema 1. A tabela a seguir descreve as conversões disponíveis suportadas pelo Amazon ECR quando uma imagem é extraída por tag:

Schema solicitado pelo cliente	Enviado ao ECR como V2, schema 1	Enviado ao ECR como V2, schema 2	Enviado ao ECR como OCI
V2, schema 1	Não é necessário converter	Convertido em V2, schema 1	Convertido em V2, schema 1
V2, schema 2	Não há conversões disponíveis, o cliente volta para V2, schema 1	Não é necessário converter	Convertido em V2, schema 2
OCI	Não há conversões disponíveis	Convertido em OCI	Não é necessário converter

Important

Se você extrair uma imagem por resumo, não há tradução disponível. O cliente precisa entender o formato do manifesto de imagem armazenado no Amazon ECR. Se você solicitar uma imagem do schema 2 de manifesto V2 de imagem de Docker por resumo em um cliente do Docker 1.9 ou anterior, a extração da imagem falhará. Para obter mais informações, consulte Compatibilidade de registro na documentação do Docker.
Neste exemplo, se você solicitar a mesma imagem por tag, o Amazon ECR converte o manifesto da imagem em um formato que o cliente possa entender. A extração da imagem é bem-sucedida.

Uso de imagens do Amazon ECR com o Amazon ECS

Você pode usar seus repositórios privados do Amazon ECR para hospedar imagens e artefatos de contêiner dos quais suas tarefas do Amazon ECS podem ser extraídas. Para que isso funcione, o agente de contêiner do Amazon ECS ou do Fargate deve ter permissões para criar as APIs `ecr:BatchGetImage`, `ecr:GetDownloadUrlForLayer` e `ecr:GetAuthorizationToken`.

Permissões obrigatórias do IAM

A tabela a seguir mostra o perfil do IAM a ser usado, para cada tipo de inicialização, que fornece as permissões necessárias para que suas tarefas sejam extraídas de um repositório privado do Amazon ECR. O Amazon ECS fornece políticas do IAM gerenciadas que incluem as permissões necessárias.

Tipo de inicialização	IAM role (Perfil do IAM)	AWS política de IAM gerenciada
Amazon ECS em instâncias do Amazon EC2	Use o perfil do IAM de instância de contêiner associado à instância do Amazon EC2 registrada em seu cluster do Amazon ECS. Para obter mais informaçõ es, consulte Perfil do IAM de instância de contêiner no Guia do desenvolvedor do Amazon Elastic Container Service.	`AmazonEC2Container ServiceforEC2Role` Para obter mais informaçõ es, consulte AmazonEC2 ContainerServiceforEC2Role no Guia do desenvolvedor do Amazon Elastic Container Service
Amazon ECS no Fargate	Use o perfil do IAM de execução de tarefas que você referencia em sua definição de tarefa do Amazon ECS. Para obter mais informações, consulte Perfil do IAM para execução de tarefa no Guia do desenvolvedor do Amazon Elastic Container Service.	`AmazonECSTaskExecu tionRolePolicy` Para obter mais informaçõ es, consulte AmazonECS TaskExecutionRolePolicy no Guia do desenvolvedor do Amazon Elastic Container Service.
Amazon ECS em instâncias externas	Use o perfil do IAM de instância de contêiner que está associado ao servidor on-premises ou à máquina virtual (VM) registrada no cluster do Amazon ECS. Para obter mais informações, consulte	`AmazonEC2Container ServiceforEC2Role` Para obter mais informaçõ es, consulte AmazonEC2 ContainerServiceforEC2Role no Guia do desenvolvedor

Tipo de inicialização	IAM role (Perfil do IAM)	AWS política de IAM gerenciada
	Perfil do Amazon ECS de instância de contêiner no Guia do desenvolvedor do Amazon Elastic Container Service.	do Amazon Elastic Container Service.

Important

As políticas AWS gerenciadas do IAM contêm permissões adicionais que talvez você não precise para seu uso. Nesse caso, estas são as permissões mínimas necessárias para extrair de um repositório privado do Amazon ECR.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer",
                "ecr:GetAuthorizationToken"
            ],
            "Resource": "*"
        }
    ]
}
```

Especificar uma imagem do Amazon ECR em uma definição de tarefa do Amazon ECS

Ao criar uma definição de tarefa do Amazon ECS, é possível especificar uma imagem de contêiner hospedada em um repositório privado do Amazon ECR. Na definição de tarefa, verifique se você está usando a nomenclatura completa de `registry/repository:tag` para as imagens do Amazon ECR. Por exemplo, *`aws_account_id`*`.dkr.ecr.`*`region`*`.amazonaws.com/`*`my-repository`*`:`*`latest`*.

O trecho de definição de tarefa a seguir mostra a sintaxe a ser usada para especificar uma imagem de contêiner hospedada no Amazon ECR em sua definição de tarefa do Amazon ECS.

```
{
    "family": "task-definition-name",
    ...
    "containerDefinitions": [
        {
            "name": "container-name",
            "image": "aws_account_id.dkr.ecr.region.amazonaws.com/my-
repository:latest",
            ...
        }
    ],
    ...
}
```

Uso de imagens do Amazon ECR com o Amazon EKS

Você pode usar suas imagens do Amazon ECR com o Amazon EKS.

Ao fazer referência a uma imagem do Amazon ECR, você deverá usar a nomenclatura de `registry/repository:tag` completa da imagem. Por exemplo, *aws_account_id*`.dkr.ecr.`*region*`.amazonaws.com/`*my-repository*`:`*latest*.

Permissões obrigatórias do IAM

Se você tiver cargas de trabalho do Amazon EKS hospedadas em nós gerenciados, nós autogerenciados ou AWS Fargate, analise o seguinte:

- Cargas de trabalho do Amazon EKS hospedadas em nós gerenciados ou autogerenciados: a função IAM (`NodeInstanceRole`) do nó de trabalho do Amazon EKS é obrigatória. A função do IAM do nó de processamento do Amazon EKS deve conter as seguintes permissões de política do IAM para o Amazon ECR.

  ```
  {
      "Version": "2012-10-17",
      "Statement": [
          {
              "Effect": "Allow",
  ```

```
            "Action": [
                "ecr:BatchCheckLayerAvailability",
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer",
                "ecr:GetAuthorizationToken"
            ],
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> Se você usou `eksctl` os AWS CloudFormation modelos em Getting Started with Amazon EKS para criar seu cluster e grupos de nós de trabalho, essas permissões do IAM são aplicadas à sua função do IAM do nó de trabalho por padrão.

- Cargas de trabalho do Amazon EKS hospedadas em AWS Fargate: Use a função de execução de pods do Fargate, que fornece aos pods permissão para extrair imagens de repositórios privados do Amazon ECR. Para obter mais informações, consulte Criar uma função de execução do pod do Fargate.

Instalação de um gráfico do Helm em um cluster Amazon EKS

Os gráficos do Helm hospedados no Amazon ECR podem ser instalados em seus clusters do Amazon EKS.

Pré-requisitos

- Use a versão mais recente do cliente do Helm. Estas etapas foram escritas usando a versão `3.9.0` do Helm. Para obter mais informações, consulte Instalação do Helm.
- Você tem pelo menos a versão `1.23.9` ou `2.6.3` da AWS CLI instalada em seu computador. Para obter mais informações, consulte Instalar ou atualizar a versão mais recente da AWS CLI.
- Você enviou um chart do Helm para o seu repositório do Amazon ECR. Para ter mais informações, consulte Enviando um gráfico do Helm para um repositório privado do Amazon ECR.
- Você configurou `kubectl` para trabalhar com o Amazon EKS. Para obter mais informações, consulte Criar um `kubeconfig` para o Amazon EKS no Manual do usuário do Amazon EKS. Se os comados a seguir forem bem-sucedidos para o cluster, a configuração estará correta.

```
kubectl get svc
```

Para instalar um gráfico do Helm em um cluster Amazon EKS

1. Autentique o cliente do Helm para o registro do Amazon ECR no qual o chart do Helm está hospedado. Os tokens de autenticação devem ser obtidos para cada registro usado e são válidos por 12 horas. Para ter mais informações, consulte Autenticação de registro privado no Amazon ECR.

   ```
   aws ecr get-login-password \
        --region us-west-2 | helm registry login \
        --username AWS \
        --password-stdin aws_account_id.dkr.ecr.region.amazonaws.com
   ```

2. Instale o chart. *helm-test-chart*Substitua pelo seu repositório e *0.1.0* pela tag do gráfico do Helm.

   ```
   helm install ecr-chart-demo oci://aws_account_id.dkr.ecr.region.amazonaws.com/helm-test-chart --version 0.1.0
   ```

   A saída deve ser semelhante a esta:

   ```
   NAME: ecr-chart-demo
   LAST DEPLOYED: Tue May 31 17:38:56 2022
   NAMESPACE: default
   STATUS: deployed
   REVISION: 1
   TEST SUITE: None
   ```

3. Verifique a instalação do chart.

   ```
   helm list -n default
   ```

   Resultado do exemplo:

   ```
   NAME            NAMESPACE       REVISION        UPDATED
        STATUS          CHART                   APP VERSION
   ```

```
ecr-chart-demo  default         1               2022-06-01 15:56:40.128669157 +0000
 UTC deployed        helm-test-chart-0.1.0   1.16.0
```

4. (Opcional) Consulte o chart do Helm instalado `ConfigMap`.

```
kubectl describe configmap helm-test-chart-configmap
```

5. Ao concluir, você pode remover a versão do chart do seu cluster.

```
helm uninstall ecr-chart-demo
```

Digitalize imagens em busca de vulnerabilidades de software no Amazon ECR

O recurso de escaneamento básico aprimorado está na versão prévia do Amazon ECR e está sujeito a alterações. Durante essa prévia pública, você só pode usar o AWS Management Console para optar pela versão de digitalização básica aprimorada.

O escaneamento de imagens do Amazon ECR ajuda a identificar vulnerabilidades de software nas imagens do seu contêiner. Os seguintes tipos de verificação são oferecidos.

Important

Alternar entre as versões de escaneamento avançado, escaneamento básico e escaneamento básico aprimorado fará com que os escaneamentos previamente estabelecidos não estejam mais disponíveis. Você precisará configurar suas digitalizações novamente. No entanto, se você voltar para a versão de escaneamento anterior, os escaneamentos estabelecidos estarão disponíveis.

- Verificação avançada: o Amazon ECR se integra ao Amazon Inspector para fornecer a verificação automatizada e contínua de seus repositórios. Suas imagens de contêiner são verificadas quanto a vulnerabilidades em sistemas operacionais e pacotes de idiomas de programação. À medida que novas vulnerabilidades aparecem, os resultados da verificação são atualizados e o Amazon Inspector emite um evento EventBridge para notificá-lo. O escaneamento aprimorado fornece o seguinte:
  - Vulnerabilidades de pacotes de sistemas operacionais e linguagens de programação.
  - Duas frequências de varredura: digitalização por push e varredura contínua.
- Escaneamento básico — O Amazon ECR fornece duas versões do escaneamento básico que usam o banco de dados Common Vulnerabilities and Exposures (CVEs): a versão atual do GA que usa o projeto Clair de código aberto e uma versão recém-aprimorada do escaneamento básico (em versão prévia) que usa nossa tecnologia nativa. AWS Com a verificação básica, você pode configurar seus repositórios para verificação durante o envio ou executar verificações manuais, e o Amazon ECR fornece uma lista de descobertas da verificação. O escaneamento básico fornece o seguinte:

- Escaneamentos do sistema operacional.
- Duas frequências de digitalização: manual e digitalização por push.

> **Important**
>
> A nova versão do escaneamento básico não suporta `imageScanFindingsSummary` e `imageScanStatus` está na `DescribeImages` API. Para visualizá-los, use a `DescribeImageScanFindings` API.

Filtros para escolher quais repositórios são escaneados no Amazon ECR

Ao configurar a digitalização de imagens para seu registro privado, você pode usar filtros para escolher quais repositórios serão digitalizados.

Quando a digitalização básica é usada, você pode especificar digitalização em filtros push para especificar quais repositórios estão definidos para fazer uma digitalização de imagem quando novas imagens forem enviadas. Quaisquer repositórios que não correspondam a uma digitalização básica no filtro push serão definidos para a frequência manual de digitalização, o que significa que, para realizar uma digitalização, você deve acioná-la manualmente.

Quando a digitalização aprimorada é usada, você pode especificar filtros separados para digitalização em push e digitalização contínua. Quaisquer repositórios que não correspondam a um filtro de digitalização aprimorada terão a digitalização desativada. Se você estiver usando a digitalização aprimorada e especificar filtros separados para digitalização em push e digitalização contínua, onde vários filtros correspondem ao mesmo repositório, o Amazon ECR impõe o filtro de digitalização contínua em vez da digitalização no filtro push para esse repositório.

Filtrar curingas

Quando um filtro é especificado, um filtro sem curinga corresponderá a todos os nomes de repositórios que contêm o filtro. Um filtro com um curinga (*) corresponde a qualquer nome de repositório em que o curinga substitui zero ou mais caracteres no nome do repositório.

A tabela a seguir fornece exemplos em que os nomes de repositórios podem ser visualizados no eixo horizontal e os filtros de exemplo são especificados no eixo vertical.

	prod	repo-prod	prod-repo	repo-prod-repo	prodrepo
prod	Sim	Sim	Sim	Sim	Sim
*prod	Sim	Sim	Não	Não	Não
prod*	Sim	Não	Sim	Não	Sim
prod	Sim	Sim	Sim	Sim	Sim
prod*repo	Não	Não	Sim	Não	Sim

Digitalize imagens em busca de vulnerabilidades de pacotes de sistemas operacionais e linguagens de programação no Amazon ECR

A verificação avançada do Amazon ECR é uma integração com o Amazon Inspector que permite a verificação de vulnerabilidades para suas imagens de contêiner. Suas imagens de contêiner são verificadas quanto a vulnerabilidades em sistemas operacionais e pacotes de linguagem de programação. Você pode ver as descobertas da verificação com o Amazon ECR e com o Amazon Inspector diretamente. Para obter mais informações sobre o Amazon Inspector, consulte Verificação de imagens de contêiner com o Amazon Inspector no Guia do usuário do Amazon Inspector.

Com a verificação avançada, você pode escolher quais repositórios são configurados para verificação automática e contínua e quais são configurados para verificação ao enviar. Isso é feito com a configuração de filtros de verificação.

Considerações sobre a verificação avançada

Considere o seguinte antes de ativar o escaneamento aprimorado do Amazon ECR.

- Não há custo adicional do Amazon ECR para usar esse recurso, no entanto, há um custo do Amazon Inspector para a digitalização das suas imagens. Para obter mais informações, consulte a Definição de preço do Amazon Inspector.
- A verificação aprimorada não é aceita nas seguintes regiões:
  - Oriente Médio (Emirados Árabes Unidos) (`me-central-1`)

  - Asia Pacific (Hyderabad) (`ap-south-2`)
  - Israel (Tel Aviv) (`il-central-1`)
  - Ásia-Pacífico (Melbourne) (`ap-southeast-4`)
  - Europa (Espanha) (`eu-south-2`)
- O Amazon Inspector oferece suporte à verificação para sistemas operacionais específicos. Para obter uma lista completa, consulte Sistemas operacionais compatíveis: verificação do Amazon ECR no Guia do usuário do Amazon Inspector.
- O Amazon Inspector usa uma função do IAM vinculada ao serviço que fornece as permissões necessárias para disponibilizar a verificação avançada para seus repositórios. O perfil do IAM vinculado ao serviço é criado automaticamente pelo Amazon Inspector quando a verificação avançada é ativada para seu registro privado. Para obter mais informações, consulte Usar funções vinculadas ao serviço do Amazon Inspector no Guia do usuário do Amazon Inspector.
- Quando você ativa inicialmente a digitalização aprimorada para seu registro privado, o Amazon Inspector reconhece apenas imagens enviadas para o Amazon ECR nos últimos 30 dias, com base no timestamp de envio da imagem, ou extraídas nos últimos 90 dias. Imagens mais antigas terão o status de verificação `SCAN_ELIGIBILITY_EXPIRED`. Se desejar que essas imagens sejam verificadas pelo Amazon Inspector, você deverá enviá-las novamente para o seu repositório.
- Todas as imagens enviadas para o Amazon ECR após a ativação da verificação aprimorada são verificadas continuamente durante o período configurado. Por padrão, a duração é Ilimitada. Essa configuração pode ser ajustada no console do Amazon Inspector. Para ter mais informações, consulte Alterando a duração aprimorada da digitalização de imagens no Amazon Inspector.
- Quando a verificação avançada está ativada no registro privado do Amazon ECR, todos os repositórios que correspondem aos filtros de verificação são verificados usando somente a verificação avançada. Todos os repositórios que não corresponderem a um filtro terão uma frequência de digitalização `Off`, mas não serão verificados. Não há suporte a digitalizações manuais com a digitalização avançada. Para ter mais informações, consulte Filtros para escolher quais repositórios são escaneados no Amazon ECR.
- Se você especificar filtros separados para digitalização em push e digitalização contínua, onde vários filtros correspondem ao mesmo repositório, o Amazon ECR impõe o filtro de digitalização contínua em vez da digitalização no filtro push para esse repositório.
- Quando a verificação aprimorada é ativada, o Amazon ECR envia um evento para EventBridge quando a frequência de varredura de um repositório é alterada. O Amazon Inspector emite eventos para EventBridge quando uma varredura inicial é concluída e quando uma descoberta de digitalização de imagem é criada, atualizada ou fechada.

Permissões do IAM necessárias para escaneamento aprimorado no Amazon ECR

A verificação avançada do Amazon ECR requer uma função do IAM vinculada ao serviço do Amazon Inspector e que a entidade principal do IAM que está habilitando e usando a verificação avançada tenha permissões para chamar as APIs do Amazon Inspector necessárias para a verificação. O perfil do IAM vinculado ao serviço do Amazon Inspector é criado automaticamente pelo Amazon Inspector quando a verificação avançada é ativada para seu registro privado. Para obter mais informações, consulte Usar funções vinculadas ao serviço do Amazon Inspector no Guia do usuário do Amazon Inspector.

A política do IAM a seguir concede as permissões necessárias para habilitar e usar a verificação avançada. Ela inclui a permissão necessária para que o Amazon Inspector crie o perfil do IAM vinculado ao serviço, bem como as permissões da API do Amazon Inspector necessárias para ativar e desativar a verificação avançada e recuperar as descobertas da verificação.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "inspector2:Enable",
                "inspector2:Disable",
                "inspector2:ListFindings",
                "inspector2:ListAccountPermissions",
                "inspector2:ListCoverage"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:AWSServiceName": [
                        "inspector2.amazonaws.com"
                    ]
                }
            }
```

```
        }
    ]
}
```

Configurando a digitalização aprimorada de imagens no Amazon ECR

Configure o escaneamento aprimorado por região para seu registro privado.

Verifique se você tem as permissões adequadas do IAM para configurar o escaneamento aprimorado. Para mais informações, consulte Permissões do IAM necessárias para escaneamento aprimorado no Amazon ECR.

AWS Management Console

Para ativar a verificação aprimorada para seu registro privado

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região para a qual deseja definir a configuração de verificação.
3. No painel de navegação, escolha Registro privado, Configurações, Digitalização.
4. Na página Scanning configuration (Configuração da verificação), em Scan type (Tipo de verificação), escolha Enhanced scanning (Verificação avançada).

   Por padrão, quando o Escaneamento avançado é selecionado, todos os seus repositórios são escaneados continuamente.
5. Para escolher repositórios específicos para verificar continuamente, desmarque a caixa Verificar continuamente todos os repositórios e defina seus filtros:

   > **Important**
   >
   > Um filtro sem curinga corresponderá a todos os nomes de repositórios que contêm o filtro. Filtros com curingas (*) correspondem a qualquer nome de repositório em que o curinga substitui zero ou mais caracteres no nome do repositório. Para ver exemplos de como os filtros se comportam, consultethe section called “Filtrar curingas”.

   a. Insira um filtro com base nos nomes dos repositórios e escolha Adicionar filtro.

b. Decida quais repositórios digitalizar quando uma imagem for enviada:

- Para verificar todos os repositórios por push, selecione Verificar por push todos os repositórios.
- Para escolher repositórios específicos para verificar por push, insira um filtro com base nos nomes dos repositórios e escolha Adicionar filtro.

6. Selecione Save (Salvar).
7. Repita estas etapas em cada região na qual deseja ativar a verificação avançada.

AWS CLI

Use o AWS CLI comando a seguir para ativar a verificação aprimorada do seu registro privado usando AWS CLI o. Você pode especificar filtros de verificação usando o objeto `rules`.

- put-registry-scanning-configuration (AWS CLI)

  O exemplo a seguir ativa a verificação avançada para seu registro privado. Por padrão, quando `rules` não são especificadas, o Amazon ECR define a configuração de verificação para verificação contínua para todos os repositórios.

```
aws ecr put-registry-scanning-configuration \
     --scan-type ENHANCED \
     --region us-east-2
```

  O exemplo a seguir ativa a verificação avançada para seu registro privado e especifica um filtro de verificação. O filtro de verificação no exemplo ativa a verificação contínua para todos os repositórios com `prod` em seu nome.

```
aws ecr put-registry-scanning-configuration \
     --scan-type ENHANCED \
     --rules '[{"repositoryFilters" : [{"filter":"prod","filterType" :
 "WILDCARD"}],"scanFrequency" : "CONTINUOUS_SCAN"}]' \
     --region us-east-2
```

  O exemplo a seguir ativa a verificação avançada para seu registro privado e especifica vários filtros de verificação. Os filtros de verificação no exemplo ativam a verificação contínua para todos os repositórios com `prod` em seu nome e ativam a verificação somente ao enviar para todos os demais repositórios.

```
aws ecr put-registry-scanning-configuration \
     --scan-type ENHANCED \
     --rules '[{"repositoryFilters" : [{"filter":"prod","filterType" :
 "WILDCARD"}],"scanFrequency" : "CONTINUOUS_SCAN"},{"repositoryFilters" :
 [{"filter":"*","filterType" : "WILDCARD"}],"scanFrequency" : "SCAN_ON_PUSH"}]' \
     --region us-west-2
```

Alterando a duração aprimorada da digitalização de imagens no Amazon Inspector

Você pode alterar o número de dias em que o Amazon Inspector digitaliza continuamente as imagens em seus repositórios privados do Amazon ECR. Por padrão, quando a verificação avançada está ativada para seu registro privado do Amazon ECR, o serviço Amazon Inspector monitora continuamente seus repositórios até que a imagem seja excluída ou a verificação avançada seja desativada. A duração pela qual o Amazon Inspector verifica suas imagens pode ser alterada usando as configurações do Amazon Inspector. As durações de verificação disponíveis são Lifetime (default) (Vitalícia (padrão)), 180 dias e 30 dias. Quando a duração da verificação de um repositório terminar, o status da verificação `SCAN_ELIGIBILITY_EXPIRED` é exibido ao listar suas verificações de vulnerabilidades. Para obter mais informações, consulte Alteração da duração da nova verificação automatizada do Amazon ECR no Guia do usuário do Amazon Inspector.

Para alterar a configuração de duração da verificação aprimorada

1. Abra o console do Amazon Inspector em https://console.aws.amazon.com/inspector/v2/home.
2. Na navegação à esquerda, expanda Settings (Configurações) e, depois, escolha General (Geral).
3. Na página Settings (Configurações), abaixo de ECR re-scan duration (Duração da nova verificação do ECR), escolha uma configuração e, em seguida, escolha Save (Salvar).

EventBridge eventos enviados para digitalização aprimorada no Amazon ECR

Quando a verificação aprimorada é ativada, o Amazon ECR envia um evento para EventBridge quando a frequência de varredura de um repositório é alterada. O Amazon Inspector envia

eventos para EventBridge quando uma varredura inicial é concluída e quando uma descoberta de digitalização de imagem é criada, atualizada ou fechada.

Evento para uma alteração de frequência de verificação do repositório

Quando a verificação avançada está ativada para seu registro, o evento a seguir é enviado pelo Amazon ECR quando há uma alteração em um recurso que tem a verificação avançada ativada. Isso inclui novos repositórios sendo criados, a frequência de verificação de um repositório sendo alterada ou quando as imagens são criadas ou excluídas em repositórios com a verificação avançada ativada. Para ter mais informações, consulte Digitalize imagens em busca de vulnerabilidades de software no Amazon ECR.


```
{
 "version": "0",
 "id": "0c18352a-a4d4-6853-ef53-0abEXAMPLE",
 "detail-type": "ECR Scan Resource Change",
 "source": "aws.ecr",
 "account": "123456789012",
 "time": "2021-10-14T20:53:46Z",
 "region": "us-east-1",
 "resources": [],
 "detail": {
  "action-type": "SCAN_FREQUENCY_CHANGE",
  "repositories": [{
    "repository-name": "repository-1",
    "repository-arn": "arn:aws:ecr:us-east-1:123456789012:repository/repository-1",
    "scan-frequency": "SCAN_ON_PUSH",
    "previous-scan-frequency": "MANUAL"
   },
   {
    "repository-name": "repository-2",
    "repository-arn": "arn:aws:ecr:us-east-1:123456789012:repository/repository-2",
    "scan-frequency": "CONTINUOUS_SCAN",
    "previous-scan-frequency": "SCAN_ON_PUSH"
   },
   {
    "repository-name": "repository-3",
    "repository-arn": "arn:aws:ecr:us-east-1:123456789012:repository/repository-3",
    "scan-frequency": "CONTINUOUS_SCAN",
    "previous-scan-frequency": "SCAN_ON_PUSH"
   }
  ],
  "resource-type": "REPOSITORY",
```

```
  "scan-type": "ENHANCED"
 }
}
```

Evento para uma verificação de imagem inicial (verificação avançada)

Quando a verificação avançada está ativada para seu registro, o evento a seguir é enviado pelo Amazon Inspector quando a verificação de imagem inicial é concluída. O parâmetro `finding-severity-counts` só retornará um valor de um nível de gravidade se existir algum. Por exemplo, se a imagem não contiver descobertas no nível `CRITICAL`, não será retornada uma contagem crítica. Para ter mais informações, consulte Digitalize imagens em busca de vulnerabilidades de pacotes de sistemas operacionais e linguagens de programação no Amazon ECR.

Padrão de evento:

```
{
  "source": ["aws.inspector2"],
  "detail-type": ["Inspector2 Scan"]
}
```

Resultado do exemplo:

```
{
    "version": "0",
    "id": "739c0d3c-4f02-85c7-5a88-94a9EXAMPLE",
    "detail-type": "Inspector2 Scan",
    "source": "aws.inspector2",
    "account": "123456789012",
    "time": "2021-12-03T18:03:16Z",
    "region": "us-east-2",
    "resources": [
        "arn:aws:ecr:us-east-2:123456789012:repository/amazon/amazon-ecs-sample"
    ],
    "detail": {
        "scan-status": "INITIAL_SCAN_COMPLETE",
        "repository-name": "arn:aws:ecr:us-east-2:123456789012:repository/amazon/
amazon-ecs-sample",
        "finding-severity-counts": {
            "CRITICAL": 7,
            "HIGH": 61,
            "MEDIUM": 62,
            "TOTAL": 158
```

```
        },
        "image-digest":
 "sha256:36c7b282abd0186e01419f2e58743e1bf635808231049bbc9d77e5EXAMPLE",
        "image-tags": [
            "latest"
        ]
    }
}
```

Evento para uma atualização de descoberta de verificação de imagem (verificação avançada)

Quando a verificação avançada está ativada para seu registro, o evento a seguir é enviado pelo Amazon Inspector quando a descoberta da verificação de imagem é criada, atualizada ou fechada. Para ter mais informações, consulte Digitalize imagens em busca de vulnerabilidades de pacotes de sistemas operacionais e linguagens de programação no Amazon ECR.

Padrão de evento:

```
{
  "source": ["aws.inspector2"],
  "detail-type": ["Inspector2 Finding"]
}
```

Resultado do exemplo:

```
{
    "version": "0",
    "id": "42dbea55-45ad-b2b4-87a8-afaEXAMPLE",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "123456789012",
    "time": "2021-12-03T18:02:30Z",
    "region": "us-east-2",
    "resources": [
        "arn:aws:ecr:us-east-2:123456789012:repository/amazon/amazon-ecs-sample/
sha256:36c7b282abd0186e01419f2e58743e1bf635808231049bbc9d77eEXAMPLE"
    ],
    "detail": {
        "awsAccountId": "123456789012",
        "description": "In libssh2 v1.9.0 and earlier versions, the SSH_MSG_DISCONNECT
 logic in packet.c has an integer overflow in a bounds check, enabling an attacker to
 specify an arbitrary (out-of-bounds) offset for a subsequent memory read. A crafted
```

```
 SSH server may be able to disclose sensitive information or cause a denial of service
 condition on the client system when a user connects to the server.",
        "findingArn": "arn:aws:inspector2:us-east-2:123456789012:finding/
be674aaddd0f75ac632055EXAMPLE",
        "firstObservedAt": "Dec 3, 2021, 6:02:30 PM",
        "inspectorScore": 6.5,
        "inspectorScoreDetails": {
            "adjustedCvss": {
                "adjustments": [],
                "cvssSource": "REDHAT_CVE",
                "score": 6.5,
                "scoreSource": "REDHAT_CVE",
                "scoringVector": "CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:N/A:N",
                "version": "3.0"
            }
        },
        "lastObservedAt": "Dec 3, 2021, 6:02:30 PM",
        "packageVulnerabilityDetails": {
            "cvss": [
                {
                    "baseScore": 6.5,
                    "scoringVector": "CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:N/A:N",
                    "source": "REDHAT_CVE",
                    "version": "3.0"
                },
                {
                    "baseScore": 5.8,
                    "scoringVector": "AV:N/AC:M/Au:N/C:P/I:N/A:P",
                    "source": "NVD",
                    "version": "2.0"
                },
                {
                    "baseScore": 8.1,
                    "scoringVector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:N/A:H",
                    "source": "NVD",
                    "version": "3.1"
                }
            ],
            "referenceUrls": [
                "https://access.redhat.com/errata/RHSA-2020:3915"
            ],
            "source": "REDHAT_CVE",
            "sourceUrl": "https://access.redhat.com/security/cve/CVE-2019-17498",
            "vendorCreatedAt": "Oct 16, 2019, 12:00:00 AM",
```

```
            "vendorSeverity": "Moderate",
            "vulnerabilityId": "CVE-2019-17498",
            "vulnerablePackages": [
                {
                    "arch": "X86_64",
                    "epoch": 0,
                    "name": "libssh2",
                    "packageManager": "OS",
                    "release": "12.amzn2.2",
                    "sourceLayerHash":
 "sha256:72d97abdfae3b3c933ff41e39779cc72853d7bd9dc1e4800c5294dEXAMPLE",
                    "version": "1.4.3"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "Update all packages in the vulnerable packages section to
 their latest versions."
            }
        },
        "resources": [
            {
                "details": {
                    "awsEcrContainerImage": {
                        "architecture": "amd64",
                        "imageHash":
 "sha256:36c7b282abd0186e01419f2e58743e1bf635808231049bbc9d77e5EXAMPLE",
                        "imageTags": [
                            "latest"
                        ],
                        "platform": "AMAZON_LINUX_2",
                        "pushedAt": "Dec 3, 2021, 6:02:13 PM",
                        "registry": "123456789012",
                        "repositoryName": "amazon/amazon-ecs-sample"
                    }
                },
                "id": "arn:aws:ecr:us-east-2:123456789012:repository/amazon/amazon-ecs-
sample/sha256:36c7b282abd0186e01419f2e58743e1bf635808231049bbc9d77EXAMPLE",
                "partition": "N/A",
                "region": "N/A",
                "type": "AWS_ECR_CONTAINER_IMAGE"
            }
        ],
```

```
        "severity": "MEDIUM",
        "status": "ACTIVE",
        "title": "CVE-2019-17498 - libssh2",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Dec 3, 2021, 6:02:30 PM"
    }
}
```

Recuperando as descobertas para escaneamentos aprimorados no Amazon ECR

Você pode recuperar os resultados do escaneamento da última digitalização aprimorada de imagem concluída e, em seguida, abrir os resultados no Amazon Inspector para ver mais detalhes. As vulnerabilidades de software que foram descobertas são listadas por gravidade com base no banco de dados Common Vulnerabilities and Exposures (CVEs).

Para obter detalhes de solução de problemas para alguns problemas comuns ao digitalizar imagens, consulte Solução de problemas de digitalização de imagens no Amazon ECR.

AWS Management Console

Use as etapas a seguir para recuperar as descobertas da verificação de imagem usando o AWS Management Console.

Para recuperar os resultados da digitalização de imagens

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região em que seu repositório reside.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositories (Repositórios), escolha o repositório que contém a imagem para a qual recuperar as descobertas da verificação.
5. Na página Images (Imagens), na coluna Vulnerabilities (Vulnerabilidades), selecione See findings (Ver descobertas) da imagem para a qual deseja recuperar as descobertas da verificação.
6. Para ver mais detalhes no console do Amazon Inspector, escolha o nome da vulnerabilidade na coluna Nome.

AWS CLI

Use o AWS CLI comando a seguir para recuperar os resultados da digitalização de imagens usando o. AWS CLIÉ possível especificar uma imagem usando a `imageTag` ou o `imageDigest`. Ambos podem ser obtidos usando o comando list-images da CLI.

- describe-image-scan-findings (AWS CLI)

  O exemplo a seguir usa uma tag de imagem.

  ```
  aws ecr describe-image-scan-findings \
       --repository-name name \
       --image-id imageTag=tag_name \
       --region us-east-2
  ```

  O exemplo a seguir usa um resumo de imagem.

  ```
  aws ecr describe-image-scan-findings \
       --repository-name name \
       --image-id imageDigest=sha256_hash \
       --region us-east-2
  ```

Escaneie imagens em busca de vulnerabilidades do sistema operacional no Amazon ECR

O recurso de escaneamento básico aprimorado está na versão prévia do Amazon ECR e está sujeito a alterações. Durante essa prévia pública, você só pode usar o AWS Management Console para optar pela versão de digitalização básica aprimorada.

O Amazon ECR fornece duas versões de escaneamento básico que usam o banco de dados Common Vulnerabilities and Exposures (CVEs):

- A versão atual do GA que usa o projeto Clair de código aberto. Para obter mais informações sobre Clair, consulte Clair on. GitHub
- A versão recém-aprimorada da digitalização básica (em versão prévia) que usa tecnologia AWS nativa.

O Amazon ECR usa a severidade de um CVE da fonte de distribuição upstream, se disponível. Caso contrário, a pontuação do Common Vulnerability Scoring System (CVSS) é usada. A pontuação do CVSS pode ser usada para obter a classificação de gravidade de vulnerabilidade do NVD. Para obter mais informações, consulte Classificações de gravidade de vulnerabilidade do NVD.

Ambas as versões do escaneamento básico do Amazon ECR oferecem suporte a filtros para especificar quais repositórios devem ser escaneados por push. Todos os repositórios que não correspondam a um escaneamento no filtro push são configurados para a frequência de escaneamento manual, o que significa que você deve iniciar o escaneamento manualmente. Uma imagem pode ser digitalizada uma vez a cada 24 horas. As 24 horas incluem a verificação inicial por push, se configurada, e todas as verificações manuais.

As últimas descobertas da verificação de imagem concluídas podem ser recuperadas para cada imagem. Quando uma digitalização de imagem é concluída, o Amazon ECR envia um evento para a Amazon EventBridge. Para ter mais informações, consulte Eventos do Amazon ECR e EventBridge.

Suporte regional para escaneamento básico aprimorado

A versão aprimorada do escaneamento básico é suportada nas seguintes regiões:

- Ásia-Pacífico (Hong Kong) (`ap-east-1`)
- Europa (Estocolmo) (`eu-north-1`)
- Oriente Médio (Bahrein) (`me-south-1`)
- Ásia-Pacífico (Mumbai) (`ap-south-1`)
- Europa (Paris) (`eu-west-3`)
- AWS GovCloud (Leste dos EUA) (`us-gov-east-1`)
- África (Cidade do Cabo) (`af-south-1`)
- Ásia-Pacífico (Jacarta) (`ap-southeast-3`)
- Europa (Frankfurt) (`eu-central-1`)
- Europa (Irlanda) (`eu-west-1`)
- América do Sul (São Paulo) (`sa-east-1`)
- Leste dos EUA (Ohio) (`us-east-2`)
- AWS GovCloud (Oeste dos EUA) (`us-gov-west-1`)
- Ásia-Pacífico (Tóquio) (`ap-northeast-1`)
- Ásia-Pacífico (Seul) (`ap-northeast-2`)

- Asia Pacific (Osaka) (`ap-northeast-3`)
- Europa (Milão) (`eu-south-1`)
- Europa (Londres) (`eu-west-2`)
- Leste dos EUA (Norte da Virgínia) (`us-east-1`)
- Ásia-Pacífico (Singapura) (`ap-southeast-1`)
- Ásia-Pacífico (Sydney) (`ap-southeast-2`)
- Canadá (Central) (`ca-central-1`)
- Oeste dos EUA (N. da Califórnia) (`us-west-1`)
- Oeste dos EUA (Oregon) (`us-west-2`)
- Europa (Zurique) (`eu-central-2`)

Suporte do sistema operacional para escaneamento básico e escaneamento básico aprimorado

Como prática recomendada de segurança e para cobertura contínua, recomendamos que você continue usando as versões compatíveis de um sistema operacional. De acordo com a política do fornecedor, os sistemas operacionais descontinuados não são mais atualizados com patches e, em muitos casos, novos alertas de segurança não são mais lançados para eles. Além disso, alguns fornecedores removem os alertas e detecções de segurança existentes de seus feeds quando um sistema operacional afetado chega ao fim do suporte padrão. Depois que uma distribuição perde o suporte de seu fornecedor, o Amazon ECR pode não suportar mais a verificação de vulnerabilidades. Qualquer descoberta que o Amazon ECR gerar para um sistema operacional descontinuado deve ser usada apenas para fins informativos. Abaixo estão listados os sistemas operacionais e as versões atualmente compatíveis.

Sistema operacional	Version (Versão)
Alpine Linux (Alpino)	3.19
Alpine Linux (Alpino)	3,18
Alpine Linux (Alpino)	3.17
Alpine Linux (Alpino)	3.16

Sistema operacional	Version (Versão)
Amazon Linux 2 (AL2)	AL2
Amazon Linux 2023 (AL2023)	AL2023
CentOS Linux (CentOS)	7
Servidor Debian (Bookworm)	12
Servidor Debian (Bullseye)	11
Servidor Debian (Buster)	10
Oracle Linux (Oracle)	9
Oracle Linux (Oracle)	8
Oracle Linux (Oracle)	7
Ubuntu (lunar)	23.04
Ubuntu (Jammy)	22.04 (LTS)
Ubuntu (Focal)	20.04 (LTS)
Ubuntu (Biônico)	18.04 (ESM)
Ubuntu (Xenial)	16.04 (ESM)
Ubuntu (Confiável)	14.04 (ESM)
Red Hat Enterprise Linux (RHEL)	7
Red Hat Enterprise Linux (RHEL)	8
Red Hat Enterprise Linux (RHEL)	9

Configurando a digitalização básica aprimorada para imagens no Amazon ECR

Uma versão aprimorada do escaneamento básico do Amazon ECR já está disponível em versão prévia. A digitalização básica aprimorada usa tecnologia AWS nativa.

Configure o escaneamento básico aprimorado por região para seu repositório privado. Para obter uma lista de regiões que oferecem suporte ao escaneamento básico aprimorado, consulteSuporte regional para escaneamento básico aprimorado.

Para ativar a verificação básica aprimorada para seu registro privado

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região para a qual deseja definir a configuração de verificação.
3. No painel de navegação, escolha Private registry (Registro privado), Scanning (Verificação).
4. Na página de configuração de digitalização, em Tipo de digitalização, escolha Digitalização básica aprimorada (em pré-visualização) - nova.
5. Por padrão, todos os repositórios estão definidos como verificação Manual. Você também pode configurar a verificação ao enviar especificando Filtros de verificação ao enviar. Você pode definir a verificação ao enviar para todos os repositórios ou para repositórios individuais. Para ter mais informações, consulte Filtros para escolher quais repositórios são escaneados no Amazon ECR.

Configurando a digitalização básica para imagens no Amazon ECR

Por padrão, o Amazon ECR ativa a verificação básica para todos os registros privados. Como resultado, a menos que você tenha alterado as configurações de escaneamento em seu registro privado, não há necessidade de ativar o escaneamento básico. A digitalização básica usa o projeto Clair de código aberto.

Você pode usar as etapas a seguir para definir uma ou mais varreduras em filtros push.

Para ativar a verificação básica do seu registro privado

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.

2. Na barra de navegação, selecione a região para a qual deseja definir a configuração de verificação.
3. No painel de navegação, escolha Private registry (Registro privado), Scanning (Verificação).
4. Na página Scanning configuration (Configuração da verificação), em Scan type (Tipo de verificação), escolha Basic scanning (Verificação básica).
5. Por padrão, todos os repositórios estão definidos como verificação Manual. Você também pode configurar a verificação ao enviar especificando Filtros de verificação ao enviar. Você pode definir a verificação ao enviar para todos os repositórios ou para repositórios individuais. Para ter mais informações, consulte Filtros para escolher quais repositórios são escaneados no Amazon ECR.

Digitalização manual de uma imagem em busca de vulnerabilidades do sistema operacional no Amazon ECR

Se seus repositórios não estiverem configurados para digitalizar por push, você poderá iniciar manualmente as digitalizações de imagens. Uma imagem pode ser digitalizada uma vez a cada 24 horas. As 24 horas incluem a verificação inicial por push, se configurada, e todas as verificações manuais.

Para obter detalhes de solução de problemas para alguns problemas comuns ao digitalizar imagens, consulte Solução de problemas de digitalização de imagens no Amazon ECR.

AWS Management Console

Use as etapas a seguir para iniciar uma verificação manual de imagem usando o AWS Management Console.

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região na qual criará o seu repositório.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositories (Repositórios), escolha o repositório que contém a imagem a ser verificada.
5. Na página Images (Imagens) selecione a imagem a ser verificada e escolha Scan (Verificar).

AWS CLI

- start-image-scan (AWS CLI)

  O exemplo a seguir usa uma tag de imagem.

```
aws ecr start-image-scan --repository-name name --image-id imageTag=tag_name --
region us-east-2
```

  O exemplo a seguir usa um resumo de imagem.

```
aws ecr start-image-scan --repository-name name --image-id imageDigest=sha256_hash
 --region us-east-2
```

AWS Tools for Windows PowerShell

- Descoberta do Get-ECR () ImageScanAWS Tools for Windows PowerShell

  O exemplo a seguir usa uma tag de imagem.

```
Start-ECRImageScan -RepositoryName name -ImageId_ImageTag tag_name -Region us-
east-2 -Force
```

  O exemplo a seguir usa um resumo de imagem.

```
Start-ECRImageScan -RepositoryName name -ImageId_ImageDigest sha256_hash -
Region us-east-2 -Force
```

Recuperando as descobertas para escaneamentos básicos no Amazon ECR

Você pode recuperar os resultados da última digitalização básica de imagem concluída. As vulnerabilidades de software que foram descobertas são listadas por gravidade com base no banco de dados Common Vulnerabilities and Exposures (CVEs).

Para obter detalhes de solução de problemas para alguns problemas comuns ao digitalizar imagens, consulte Solução de problemas de digitalização de imagens no Amazon ECR.

AWS Management Console

Use as etapas a seguir para recuperar as descobertas da verificação de imagem usando o AWS Management Console.

Para recuperar os resultados da digitalização de imagens

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região na qual criará o seu repositório.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositories (Repositórios), escolha o repositório que contém a imagem para a qual recuperar as descobertas da verificação.
5. Na página Images (Imagens), na coluna Vulnerabilities (Vulnerabilidades), selecione Details (Detalhes) da imagem para a qual deseja recuperar as descobertas da verificação.

AWS CLI

Use o AWS CLI comando a seguir para recuperar os resultados da digitalização de imagens usando o. AWS CLIÉ possível especificar uma imagem usando a `imageTag` ou o `imageDigest`. Ambos podem ser obtidos usando o comando list-images da CLI.

- describe-image-scan-findings (AWS CLI)

  O exemplo a seguir usa uma tag de imagem.

  ```
  aws ecr describe-image-scan-findings --repository-name name --image-id
   imageTag=tag_name --region us-east-2
  ```

  O exemplo a seguir usa um resumo de imagem.

  ```
  aws ecr describe-image-scan-findings --repository-name name --image-id
   imageDigest=sha256_hash --region us-east-2
  ```

AWS Tools for Windows PowerShell

- Descoberta do Get-ECR () ImageScanAWS Tools for Windows PowerShell

  O exemplo a seguir usa uma tag de imagem.

```
Get-ECRImageScanFinding -RepositoryName name -ImageId_ImageTag tag_name -
Region us-east-2
```

O exemplo a seguir usa um resumo de imagem.

```
Get-ECRImageScanFinding -RepositoryName name -ImageId_ImageDigest sha256_hash -
Region us-east-2
```

Solução de problemas de digitalização de imagens no Amazon ECR

Veja a seguir falhas comuns de verificação de imagem. Você pode visualizar erros como esses no console do Amazon ECR exibindo os detalhes da imagem ou por meio da API ou AWS CLI usando a `DescribeImageScanFindings` API.

UnsupportedImageErro

Você pode receber um erro de `UnsupportedImageError` ao tentar realizar uma verificação básica em uma imagem que foi criada usando um sistema operacional para o qual o Amazon ECR não oferece suporte à verificação de imagens. O Amazon ECR suporta verificação de vulnerabilidades de pacotes para as versões principais de distribuições do Amazon Linux, Amazon Linux 2, Debian, Ubuntu, CentOS, Oracle Linux, Alpine e RHEL Linux. Quando uma distribuição perde o suporte de seu fornecedor, o Amazon ECR pode não suportar a verificação de vulnerabilidades dessa distribuição. O Amazon ECR não suporta verificação de imagens criadas a partir de imagem de scratch do Docker.

> **Important**
>
> Ao usar a verificação avançada, o Amazon Inspector oferece suporte à verificação para sistemas operacionais e tipos de mídias específicos. Para obter uma lista completa, consulte Sistemas operacionais compatíveis: verificação do Amazon ECR no Guia do usuário do Amazon Inspector.

Um nível de severidade `UNDEFINED` é retornado

Você poderá receber uma descoberta de verificação que tenha um nível de severidade `UNDEFINED`. As causas comuns para isso são as seguintes:

- A origem do CVE não atribuiu uma prioridade à vulnerabilidade.
- A vulnerabilidade recebeu uma prioridade que o Amazon ECR não reconhece.

Para determinar a gravidade e a descrição de uma vulnerabilidade, você pode exibir o CVE diretamente da origem.

Noções básicas do status de verificação **`SCAN_ELIGIBILITY_EXPIRED`**

Quando a verificação aprimorada usando o Amazon Inspector estiver habilitada para seu registro privado e você estiver visualizando suas vulnerabilidades de verificação, poderá ver um status de verificação de `SCAN_ELIGIBILITY_EXPIRED`. As causas comuns para isso são as seguintes:

- Quando você ativa inicialmente a verificação avançada para seu registro privado, o Amazon Inspector reconhece apenas imagens enviadas para o Amazon ECR nos últimos 30 dias com base na data e hora do envio da imagem. Imagens mais antigas terão o status de verificação `SCAN_ELIGIBILITY_EXPIRED`. Se desejar que essas imagens sejam verificadas pelo Amazon Inspector, você deverá enviá-las novamente para o seu repositório.
- Se ECR re-scan duration (Duração da nova verificação do ECR) for alterada no console do Amazon Inspector e, quando esse tempo decorre, o status da verificação da imagem é alterado para `inactive` com um código de motivo `expired`, e todas as descobertas associadas à imagem são programadas para serem fechadas. Com isso, o console do Amazon ECR lista o status da verificação como `SCAN_ELIGIBILITY_EXPIRED`.

Sincronize um registro upstream com um registro privado do Amazon ECR

Usando regras de cache pull through, você pode sincronizar o conteúdo de um registro upstream com seu registro privado do Amazon ECR.

No momento, o Amazon ECR oferece suporte à criação de regras de cache de pull-through para os seguintes registros upstream.

- Docker Hub, Microsoft Azure Container Registry, GitHub Container Registry e GitLab Container Registry (requer autenticação)
- Amazon ECR Public, o registro de imagens de contêineres do Kubernetes e o Quay (não requer autenticação)

Para o GitLab Container Registry, o Amazon ECR suporta pull through cache somente com a GitLab software-as-a-service oferta, GitLab .com.

Para os registros upstream que exigem autenticação, você deve armazenar suas credenciais em segredo. AWS Secrets Manager O console do Amazon ECR facilita a criação do segredo do Secrets Manager para cada um dos registros upstream autenticados. Para obter mais informações sobre como criar um segredo do Secrets Manager usando o console do Secrets Manager, consulteArmazenando suas credenciais do repositório upstream em segredo AWS Secrets Manager.

Após uma regra de cache de pull-through para o registro upstream, basta extrair uma imagem desse registro upstream usando o URI de registro privado do Amazon ECR. Em seguida, o Amazon ECR cria um repositório e armazena essa imagem em cache no seu registro privado. Em suas pull requests subsequentes da imagem em cache com uma determinada tag, o Amazon ECR verifica o registro upstream para ver se há uma nova versão da imagem com essa tag específica e tenta atualizar a imagem em seu registro privado pelo menos uma vez a cada 24 horas.

Modelos de criação de repositórios

O Amazon ECR adicionou suporte para modelos de criação de repositórios, atualmente em versão prévia, o que lhe dá o controle para especificar configurações iniciais para novos repositórios criados pelo Amazon ECR em seu nome usando regras de cadhe de pull-through. Cada modelo contém um prefixo de namespace do repositório que é usado para associar novos repositórios a um modelo específico. Os modelos podem especificar a configuração para todas as configurações do

repositório, incluindo políticas de acesso baseadas em recursos, imutabilidade de tags, criptografia e políticas de ciclo de vida. As configurações em um modelo de criação de repositório são aplicadas apenas durante a criação do repositório e não têm efeito sobre repositórios existentes ou repositórios criados usando qualquer outro método. Para ter mais informações, consulte Modelos para controlar repositórios criados durante uma ação de extração do cache.

Considerações sobre o uso de regras de cache pull through

Considere o seguinte ao usar as regras de cache pull through do Amazon ECR.

- A criação de regras de cache de pull-through não é aceita nas seguintes Regiões:
  - China (Pequim) (`cn-north-1`)
  - China (Ningxia) (`cn-northwest-1`)
  - AWS GovCloud (Leste dos EUA) (`us-gov-east-1`)
  - AWS GovCloud (Oeste dos EUA) (`us-gov-west-1`)
- AWS Lambda não suporta a extração de imagens de contêineres do Amazon ECR usando uma regra de cache pull through.
- Ao extrair imagens usando o cache de pull-through, os endpoints de serviço FIPS do Amazon ECR não são suportados na primeira vez que uma imagem é extraída. No entanto, usar os endpoints de serviço FIPS do Amazon ECR funciona em extrações subsequentes.
- Quando uma imagem em cache é extraída por meio do URI de registro privado do Amazon ECR, a extração da imagem é iniciada por AWS endereços IP. Isso garante que o pull da imagem não seja contabilizado em nenhuma cota de taxa de pull implementada pelo registro upstream.
- Quando uma imagem armazenada em cache é puxada por meio do URI do registro privado da Amazon ECR, o Amazon ECR verifica o repositório upstream pelo menos uma vez a cada 24 horas para verificar se a imagem em cache é a versão mais recente. Se houver uma imagem mais recente no registro upstream, o Amazon ECR tentará atualizar a imagem em cache. Este temporizador é baseado na última extração da imagem em cache.
- Se o Amazon ECR não conseguir atualizar a imagem do registro upstream por qualquer motivo e a imagem for extraída, a última imagem em cache ainda será extraída.
- Ao criar o segredo do Secrets Manager que contém as credenciais do registro upstream, o nome do segredo deve usar o prefixo `ecr-pullthroughcache/`. O segredo também deve estar na mesma conta e região em que a regra de cache de pull-through foi criada.
- Quando uma imagem multiarquitetura é extraída por meio de uma regra de cache de pull-through, a lista de manifestos e cada imagem referenciada na lista de manifesto são extraídas para o

repositório do Amazon ECR. Se desejar apenas extrair uma arquitetura específica, você poderá extrair a imagem usando o resumo da imagem ou a tag associada à arquitetura, em vez da tag associada à lista de manifesto.

- O Amazon ECR utiliza um perfil do IAM vinculada ao serviço, que fornece as permissões necessárias para o Amazon ECR criar o repositório, recuperar o valor do segredo do Secrets Manager para autenticação e enviar a imagem armazenada em cache em seu nome. A função do IAM vinculada ao serviço é criada automaticamente quando uma regra de cache de pull-through é criada. Para ter mais informações, consulte Função vinculada ao serviço do Amazon ECR para cache de pull-through.
- Por padrão, a entidade principal do IAM que está puxando a imagem armazenada em cache tem as permissões concedidas a ele por meio de sua política do IAM. Você pode usar a política de permissões de registro privado do Amazon ECR para aumentar o escopo das permissões de uma entidade do IAM. Para ter mais informações, consulte Usar permissões de registro.
- Os repositórios do Amazon ECR criados usando o fluxo de trabalho de cache de pull-through são tratados como qualquer outro repositório do Amazon ECR. Todos os recursos do repositório, como replicação e verificação de imagens, são compatíveis.
- Quando o Amazon ECR cria um novo repositório em seu nome usando uma ação de cache de pull-through, as seguintes configurações padrão são aplicadas ao repositório, a menos que haja um modelo correspondente de criação de repositório. Você pode usar um modelo de criação de repositório para definir as configurações aplicadas aos repositórios criados pelo Amazon ECR em seu nome. Para ter mais informações, consulte Modelos para controlar repositórios criados durante uma ação de extração do cache.
  - Imutabilidade da tag — Desativada, as tags são mutáveis e podem ser sobrescritas.
  - Criptografia — A criptografia padrão do `AES256` é usada.
  - Permissões do repositório — Omitida, nenhuma política de permissões do repositório é aplicada.
  - Política de ciclo de vida — omitida, nenhuma política de ciclo de vida é aplicada.
  - Tags de recursos — Omitidas, nenhuma tag de recurso é aplicada.
- Ativar a imutabilidade da tag de imagem para repositórios usando uma regra de cache de pull-through impedirá que o Amazon ECR atualize imagens usando a mesma tag.
- Quando uma imagem é extraída usando a regra de cache pull through pela primeira vez, uma rota para a Internet pode ser necessária. Há certas circunstâncias em que uma rota para a Internet é necessária, então é melhor configurar uma rota para evitar falhas. Portanto, se você configurou o Amazon ECR para usar uma interface VPC endpoint AWS PrivateLink , precisará garantir que o primeiro pull tenha uma rota para a Internet. Uma maneira de fazer isso é criar uma sub-rede

pública na mesma VPC, com um gateway de Internet, e depois rotear todo o tráfego de saída para a Internet da sub-rede privada para a sub-rede pública. As extrações de imagens subsequentes usando a regra de cache pull through não exigem isso. Para obter mais informações, consulte Opções de rotas de exemplos no Guia do usuário da Amazon Virtual Private Cloud.

Permissões do IAM necessárias para sincronizar um registro upstream com um registro privado do Amazon ECR

Além das permissões da API do Amazon ECR necessárias para autenticar em um registro privado e enviar e extrair imagens, as seguintes permissões adicionais são necessárias para usar regras de cache de pull-through de forma efetiva.

- `ecr:CreatePullThroughCacheRule`: concede permissão para criar regra de cache de pull-through. Essa permissão deve ser concedida por meio de uma política do IAM baseada em identidade.
- `ecr:BatchImportUpstreamImage`: concede permissão para recuperar a imagem externa e importá-la para o registro privado. Essa permissão pode ser concedida usando a política de permissões do registro privado, uma política do IAM baseada em identidade ou a política de permissões de repositório baseadas em recursos. Para obter mais informações sobre o uso de permissões de repositório, consulte Políticas de repositório privado no Amazon ECR.
- `ecr:CreateRepository`: concede permissão para criar um repositório em um registro privado. Essa permissão é necessária se o repositório de armazenamento de imagens em cache ainda não existir. Essa permissão pode ser concedida por uma política do IAM baseada em identidade ou pela política de permissões do registro privado.
- `ecr:TagResource` - Concede permissão para adicionar etiquetas de metadados a um recurso do Amazon ECR. Essa permissão só é necessária se você estiver extraindo uma imagem que usa uma regra de cache de pull-through que tenha um modelo de criação de repositório associado configurado para adicionar tags de recursos ao repositório. Essa permissão deve ser concedida por meio de uma política do IAM baseada em identidade.

Usar permissões de registro

As permissões de registro privado do Amazon ECR podem ser usadas para dimensionar o escopo das permissões de entidades individuais do IAM para usar o cache de pull-through. Se uma entidade do IAM tiver mais permissões concedidas por uma política do IAM do que a política de permissões

do registro está concedendo, a política do IAM terá precedência. Por exemplo, se um usuário já tiver as permissões `ecr:*`, não serão necessárias permissões adicionais no nível do registro.

Para criar uma política de permissões de um registro privado (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/
2. Na barra de navegação, escolha a região na qual deseja configurar a sua declaração de permissões da política do registro.
3. No painel de navegação, escolha Private registry (Registro privado), Registry permissions (Permissões do registro).
4. Na página Registry permissions (Permissões do registro), escolha Generate statement (Gerar declaração).
5. Para cada declaração de política de permissões de cache de pull-through que você criar, faça o seguinte.

    a. Em Policy type (Tipo de política), escolha Pull through cache policy (Política de cache de pull-through).

    b. Em Statement id (ID da declaração), forneça um nome para a política de declaração do cache de pull-through.

    c. Em IAM entities (Entidades do IAM), especifique os usuários, grupos ou funções a serem incluídos na política.

    d. Em Repository namespace (Namespace do repositório), selecione a regra de cache de pull-through para associar à política.

    e. Em Repository names (Nomes de repositórios), especifique o nome da base do repositório ao qual a regra será aplicada. Por exemplo, se você quisesse especificar o repositório do Amazon Linux no Amazon ECR Public, o nome do repositório seria `amazonlinux`.

Para criar uma política de permissões de um registro privado (AWS CLI)

Use o AWS CLI comando a seguir para especificar as permissões do registro privado usando AWS CLI o.

1. Crie um arquivo local denominado `ptc-registry-policy.json` com o conteúdo da política do registro. O exemplo a seguir concede a permissão `ecr-pull-through-cache-user` para criar um repositório e extrair uma imagem do Amazon ECR Public, que é a fonte upstream associada à regra de cache pull-through criada anteriormente.

```
{
  "Sid": "PullThroughCacheFromReadOnlyRole",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:user/ecr-pull-through-cache-user"
  },
  "Action": [
    "ecr:CreateRepository",
    "ecr:BatchImportUpstreamImage"
  ],
  "Resource": "arn:aws:ecr:us-east-1:111122223333:repository/ecr-public/*"
}
```

> **Important**
>
> A permissão `ecr-CreateRepository` é necessária se o repositório de armazenamento de imagens em cache ainda não existir. Por exemplo, se a ação de criação do repositório e as ações de extração de imagem forem realizadas por entidades separadas do IAM, como um administrador e um desenvolvedor.

2. Use um comando put-registry-policy para definir a política do registro.

```
aws ecr put-registry-policy \
     --policy-text file://ptc-registry.policy.json
```

Próximas etapas

Quando você estiver pronto para começar a usar as regras de cache de pull-through, as próximas etapas são apresentadas a seguir.

- Crie uma regra de cache de pull-through. Para ter mais informações, consulte Criação de uma regra de cache pull through no Amazon ECR.
- Crie um modelo de criação de repositório. Um modelo de criação de repositório oferece a você o controle para definir as configurações a serem usadas para novos repositórios criados pelo Amazon ECR em seu nome durante uma ação do cache de pull-through. Para ter mais informações, consulte Modelos para controlar repositórios criados durante uma ação de extração do cache.

Criação de uma regra de cache pull through no Amazon ECR

Para cada registro upstream contendo imagens que você deseja armazenar em cache no seu registro privado do Amazon ECR, você deve criar uma regra de cache pull through.

Para registros upstream que exigem autenticação, você deve armazenar as credenciais em um segredo do Secrets Manager. Você pode usar uma senha existente do ou criar outra. Você pode criar o segredo do Secrets Manager no console do Amazon ECR ou no console do Secrets Manager. Para criar um segredo do Secrets Manager usando o console do Secrets Manager em vez do console do Amazon ECR, consulteArmazenando suas credenciais do repositório upstream em segredo AWS Secrets Manager.

Pré-requisitos

- Verifique se você tem as permissões adequadas do IAM para criar regras de cache de pull through. Para mais informações, consulte Permissões do IAM necessárias para sincronizar um registro upstream com um registro privado do Amazon ECR.
- Para registros upstream que exigem autenticação: Se você quiser usar um segredo existente, verifique se o segredo do Secrets Manager atende aos seguintes requisitos:
  - O nome do segredo começa com`ecr-pullthroughcache/`. AWS Management Console Só exibe segredos do Secrets Manager com o `ecr-pullthroughcache/` prefixo.
  - A conta e a região em que o segredo está devem corresponder à conta e à região em que a regra de cache pull through está.

Para criar uma regra de cache de pull-through (AWS Management Console)

As etapas a seguir mostram como criar uma regra do cache de pull-through e um segredo do Secrets Manager usando o console do Amazon ECR. Para criar um segredo usando o console do Secrets Manager, consulteArmazenando suas credenciais do repositório upstream em segredo AWS Secrets Manager.

Para Amazon ECR Public, Kubernetes Container Registry ou Quay

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/
2. Na barra de navegação, selecione a região para a qual deseja ajustar as configurações do registro privado.

3. No painel de navegação, escolha Private registry (Registro privado), Pull through cache (Cache de pull-through).
4. Na página Pull through cache configuration (Configuração do cache de pull-through), escolha Add rule (Adicionar regra).
5. Na Etapa 1: especificar uma página de origem, em Registro, escolha Amazon ECR Public, Kubernetes ou Quay na lista de registros upstream e, em seguida, escolha Avançar.
6. Na Etapa 2: especificar uma página de destino, para o prefixo do repositório Amazon ECR, especifique o prefixo do namespace do repositório a ser usado ao armazenar em cache imagens retiradas do registro público de origem e escolha Avançar. Um namespace é preenchido por padrão, mas também é possível especificar um namespace personalizado.
7. Na página Etapa 3: Revisar e criar, revise a configuração da regra de cache de pull-through e escolha Criar.
8. Repita a etapa anterior para cada cache de pull-through que deseja criar. As regras de cache de pull-through são criadas separadamente para cada região.

Para o Docker Hub

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/
2. Na barra de navegação, selecione a região para a qual deseja ajustar as configurações do registro privado.
3. No painel de navegação, escolha Private registry (Registro privado), Pull through cache (Cache de pull-through).
4. Na página Pull through cache configuration (Configuração do cache de pull-through), escolha Add rule (Adicionar regra).
5. Na Etapa 1: especificar uma página de origem, em Registro, escolha Docker Hub, Avançar.
6. Na página Etapa 2: configurar autenticação, para credenciais do Upstream, você deve armazenar suas credenciais de autenticação para o Docker Hub em um segredo AWS Secrets Manager . Você pode especificar um segredo existente ou usar o console do Amazon ECR para criar um novo segredo.

    a. Para usar um segredo existente, escolha Usar um AWS segredo existente. Em Nome secreto, use o menu suspenso para selecionar seu segredo existente e, em seguida, escolha Avançar.

> **Note**
>
> AWS Management Console Só exibe segredos do Secrets Manager com nomes usando o `ecr-pullthroughcache/` prefixo. O segredo também deve estar na mesma conta e região em que a regra de cache de pull-through foi criada.

b. Para criar um novo segredo, escolha Criar um AWS segredo, faça o seguinte e escolha Avançar.

   i. Em Nome secreto, especifique um nome descritivo para o segredo. Os nomes de segredos devem conter de 1 a 512 caracteres Unicode.

   ii. Para nome de usuário do Docker Hub, especifique seu nome de usuário do Docker Hub.

   iii. Para o token de acesso do Docker Hub, especifique seu token de acesso do Docker Hub. Para obter mais informações sobre como criar um token de acesso do Docker Hub, consulte Criar e gerenciar tokens de acesso na documentação do Docker.

7. Na Etapa 3: especificar uma página de destino, para o prefixo do repositório Amazon ECR, especifique o namespace do repositório a ser usado ao armazenar em cache imagens retiradas do registro público de origem e escolha Avançar.

   Um namespace é preenchido por padrão, mas também é possível especificar um namespace personalizado.

8. Na página Etapa 4: revisar e criar, revise a configuração da regra de cache de pull-through e escolha Criar.

9. Repita a etapa anterior para cada cache de pull-through que deseja criar. As regras de cache de pull-through são criadas separadamente para cada região.

Para GitHub Container Registry

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/

2. Na barra de navegação, selecione a região para a qual deseja ajustar as configurações do registro privado.

3. No painel de navegação, escolha Private registry (Registro privado), Pull through cache (Cache de pull-through).

4. Na página Pull through cache configuration (Configuração do cache de pull-through), escolha Add rule (Adicionar regra).
5. Na Etapa 1: Especificar uma página de origem, em Registro, escolha Registro de GitHub contêiner, Avançar.
6. Na página Etapa 2: Configurar autenticação, para credenciais do Upstream, você deve armazenar suas credenciais de autenticação para o GitHub Container Registry em um segredo. AWS Secrets Manager Você pode especificar um segredo existente ou usar o console do Amazon ECR para criar um novo segredo.

   a. Para usar um segredo existente, escolha Usar um AWS segredo existente. Em Nome secreto, use o menu suspenso para selecionar seu segredo existente e, em seguida, escolha Avançar.

      > **Note**
      >
      > AWS Management Console Só exibe segredos do Secrets Manager com nomes usando o `ecr-pullthroughcache/` prefixo. O segredo também deve estar na mesma conta e região em que a regra de cache de pull-through foi criada.

   b. Para criar um novo segredo, escolha Criar um AWS segredo, faça o seguinte e escolha Avançar.

      i. Em Nome secreto, especifique um nome descritivo para o segredo. Os nomes de segredos devem conter de 1 a 512 caracteres Unicode.

      ii. Para nome de usuário do GitHub Container Registry, especifique seu nome de usuário do GitHub Container Registry

      iii. Para o token de acesso do GitHub Container Registry, especifique seu token de acesso do GitHub Container Registry. Para obter mais informações sobre a criação de um token de GitHub acesso, consulte Gerenciando seus tokens de acesso pessoais na GitHub documentação.

7. Na Etapa 3: especificar uma página de destino, para o prefixo do repositório Amazon ECR, especifique o namespace do repositório a ser usado ao armazenar em cache imagens retiradas do registro público de origem e escolha Avançar.

   Um namespace é preenchido por padrão, mas também é possível especificar um namespace personalizado.

8. Na página Etapa 4: revisar e criar, revise a configuração da regra de cache de pull-through e escolha Criar.
9. Repita a etapa anterior para cada cache de pull-through que deseja criar. As regras de cache de pull-through são criadas separadamente para cada região.

Para o Registro de Contêiner do Microsoft Azure

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/
2. Na barra de navegação, selecione a região para a qual deseja ajustar as configurações do registro privado.
3. No painel de navegação, escolha Private registry (Registro privado), Pull through cache (Cache de pull-through).
4. Na página Pull through cache configuration (Configuração do cache de pull-through), escolha Add rule (Adicionar regra).
5. Na Etapa 1: especificar uma página de origem, faça o seguinte.
   a. Em Registro, escolha Microsoft Azure Container Registry
   b. Em URL do registro de origem, especifique o nome do seu registro de contêiner do Microsoft Azure e escolha Avançar.

   > **Important**
   >
   > Você só precisa especificar o prefixo, pois o sufixo `.azurecr.io` é preenchido em seu nome.

6. Na página Etapa 2: configurar autenticação, para credenciais do Upstream, você deve armazenar suas credenciais de autenticação para o Microsoft Azure Container Registry em um segredo AWS Secrets Manager . Você pode especificar um segredo existente ou usar o console do Amazon ECR para criar um novo segredo.
   a. Para usar um segredo existente, escolha Usar um AWS segredo existente. Em Nome secreto, use o menu suspenso para selecionar seu segredo existente e, em seguida, escolha Avançar.

> **Note**
>
> AWS Management Console Só exibe segredos do Secrets Manager com nomes usando o `ecr-pullthroughcache/` prefixo. O segredo também deve estar na mesma conta e região em que a regra de cache de pull-through foi criada.

b. Para criar um novo segredo, escolha Criar um AWS segredo, faça o seguinte e escolha Avançar.

   i. Em Nome secreto, especifique um nome descritivo para o segredo. Os nomes de segredos devem conter de 1 a 512 caracteres Unicode.

   ii. Para o nome de usuário do Registro de Contêiner do Microsoft Azure, especifique seu nome de usuário do Registro de Contêiner do Microsoft Azure.

   iii. Para o token de acesso do Registro de Contêiner do Microsoft Azure, especifique seu token de acesso do Registro de Contêiner do Microsoft Azure. Para obter mais informações sobre a criação de um token de acesso ao Registro de Contêiner do Microsoft Azure, consulte Criar token - portal na documentação do Microsoft Azure.

7. Na Etapa 3: especificar uma página de destino, para o prefixo do repositório Amazon ECR, especifique o namespace do repositório a ser usado ao armazenar em cache imagens retiradas do registro público de origem e escolha Avançar.

   Um namespace é preenchido por padrão, mas também é possível especificar um namespace personalizado.

8. Na página Etapa 4: revisar e criar, revise a configuração da regra de cache de pull-through e escolha Criar.

9. Repita a etapa anterior para cada cache de pull-through que deseja criar. As regras de cache de pull-through são criadas separadamente para cada região.

Para GitLab Container Registry

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/

2. Na barra de navegação, selecione a região para a qual deseja ajustar as configurações do registro privado.

3. No painel de navegação, escolha Private registry (Registro privado), Pull through cache (Cache de pull-through).

4. Na página Pull through cache configuration (Configuração do cache de pull-through), escolha Add rule (Adicionar regra).
5. Na Etapa 1: Especificar uma página de origem, em Registro, escolha Registro de GitLab contêiner, Avançar.
6. Na página Etapa 2: Configurar autenticação, para credenciais do Upstream, você deve armazenar suas credenciais de autenticação para o GitLab Container Registry em um segredo. AWS Secrets Manager Você pode especificar um segredo existente ou usar o console do Amazon ECR para criar um novo segredo.

   a. Para usar um segredo existente, escolha Usar um AWS segredo existente. Em Nome secreto, use o menu suspenso para selecionar seu segredo existente e, em seguida, escolha Avançar. Para obter mais informações sobre como criar um segredo no Secrets Manager usando o console do Secrets Manager, consulte Armazenando suas credenciais do repositório upstream em segredo AWS Secrets Manager.

   > **Note**
   >
   > AWS Management Console Só exibe segredos do Secrets Manager com nomes usando o `ecr-pullthroughcache/` prefixo. O segredo também deve estar na mesma conta e região em que a regra de cache de pull-through foi criada.

   b. Para criar um novo segredo, escolha Criar um AWS segredo, faça o seguinte e escolha Avançar.

      i. Em Nome secreto, especifique um nome descritivo para o segredo. Os nomes de segredos devem conter de 1 a 512 caracteres Unicode.

      ii. Para nome de usuário do GitLab Container Registry, especifique seu nome de usuário do GitLab Container Registry

      iii. Para o token de acesso do GitLab Container Registry, especifique seu token de acesso do GitLab Container Registry. Para obter mais informações sobre a criação de um token de acesso ao GitLab Container Registry, consulte Tokens de acesso pessoal, Tokens de acesso de grupo ou Tokens de acesso ao projeto, na GitLab documentação.
7. Na Etapa 3: especificar uma página de destino, para o prefixo do repositório Amazon ECR, especifique o namespace do repositório a ser usado ao armazenar em cache imagens retiradas do registro público de origem e escolha Avançar.

Um namespace é preenchido por padrão, mas também é possível especificar um namespace personalizado.

8. Na página Etapa 4: revisar e criar, revise a configuração da regra de cache de pull-through e escolha Criar.
9. Repita a etapa anterior para cada cache de pull-through que deseja criar. As regras de cache pull-through são criadas separadamente para cada região.

Para criar uma regra de cache de pull-through (AWS CLI)

Use o AWS CLI comando create-pull-through-cache-rule para criar uma regra de cache pull through para um registro privado do Amazon ECR. Para registros de upstream que exigem autenticação, você deve armazenar as credenciais em um segredo do Secrets Manager. Para criar um segredo usando o console do Secrets Manager, consulteArmazenando suas credenciais do repositório upstream em segredo AWS Secrets Manager.

Os exemplos a seguir são fornecidos para cada registro upstream compatível.

Para o Amazon ECR Public

O exemplo a seguir cria uma regra de cache de pull-through para o registro público do Amazon ECR. Ele especifica um prefixo de repositório de `ecr-public`, o que faz com que cada repositório criado usando a regra de cache de pull-through receba o esquema de nomes de `ecr-public/`*`upstream-repository-name`*.

```
aws ecr create-pull-through-cache-rule \
     --ecr-repository-prefix ecr-public \
     --upstream-registry-url public.ecr.aws \
     --region us-east-2
```

Para registro de contêineres Kubernetes

O exemplo a seguir cria uma regra de cache de pull-through para o registro público do Kubernetes. Ele especifica um prefixo de repositório de `kubernetes`, o que faz com que cada repositório criado usando a regra de cache de pull-through receba o esquema de nomes de `kubernetes/`*`upstream-repository-name`*.

```
aws ecr create-pull-through-cache-rule \
     --ecr-repository-prefix kubernetes \
```

```
    --upstream-registry-url registry.k8s.io \
    --region us-east-2
```

Para Quay

O exemplo a seguir cria uma regra de cache de pull-through para o registro público do Quay. Ele especifica um prefixo de repositório de `quay`, o que faz com que cada repositório criado usando a regra de cache de pull-through receba o esquema de nomes de `quay/`*`upstream-repository-name`*.

```
aws ecr create-pull-through-cache-rule \
    --ecr-repository-prefix quay \
    --upstream-registry-url quay.io \
    --region us-east-2
```

Para o Docker Hub

O exemplo a seguir cria uma regra de cache de pull-through para o registro do Docker Hub. Ele especifica um prefixo de repositório de `docker-hub`, o que faz com que cada repositório criado usando a regra de cache de pull-through receba o esquema de nomes de `docker-hub/`*`upstream-repository-name`*. É necessário especificar o nome completo do Amazon Resource Name (ARN) do segredo que contém suas credenciais do Docker Hub.

```
aws ecr create-pull-through-cache-rule \
    --ecr-repository-prefix docker-hub \
    --upstream-registry-url registry-1.docker.io \
    --credential-arn arn:aws:secretsmanager:us-east-2:111122223333:secret:ecr-
pullthroughcache/example1234 \
    --region us-east-2
```

Para GitHub Container Registry

O exemplo a seguir cria uma regra de cache pull through para o GitHub Container Registry. Ele especifica um prefixo de repositório de `docker-hub`, o que faz com que cada repositório criado usando a regra de cache de pull-through receba o esquema de nomes de `github/`*`upstream-repository-name`*. Você deve especificar o Amazon Resource Name (ARN) completo do segredo que contém suas credenciais do GitHub Container Registry.

```
aws ecr create-pull-through-cache-rule \
    --ecr-repository-prefix github \
    --upstream-registry-url ghcr.io \
```

```
     --credential-arn arn:aws:secretsmanager:us-east-2:111122223333:secret:ecr-
pullthroughcache/example1234 \
     --region us-east-2
```

Para o Registro de Contêiner do Microsoft Azure

O exemplo a seguir cria uma regra de cache pull through para o Registro de Contêiner do Microsoft Azure. Ele especifica um prefixo de repositório de `azure`, o que faz com que cada repositório criado usando a regra de cache de pull-through receba o esquema de nomes de `azure/`*`upstream-repository-name`*. É necessário especificar o nome completo do Amazon Resource Name (ARN) do segredo que contém suas credenciais do Docker Hub.

```
aws ecr create-pull-through-cache-rule \
     --ecr-repository-prefix azure \
     --upstream-registry-url myregistry.azurecr.io \
     --credential-arn arn:aws:secretsmanager:us-east-2:111122223333:secret:ecr-
pullthroughcache/example1234 \
     --region us-east-2
```

Para GitLab Container Registry

O exemplo a seguir cria uma regra de cache pull through para o GitLab Container Registry. Ele especifica um prefixo de repositório de `gitlab`, o que faz com que cada repositório criado usando a regra de cache de pull-through receba o esquema de nomes de `gitlab/`*`upstream-repository-name`*. Você deve especificar o Amazon Resource Name (ARN) completo do segredo que contém suas credenciais do GitLab Container Registry.

```
aws ecr create-pull-through-cache-rule \
     --ecr-repository-prefix gitlab \
     --upstream-registry-url registry.gitlab.com \
     --credential-arn arn:aws:secretsmanager:us-east-2:111122223333:secret:ecr-
pullthroughcache/example1234 \
     --region us-east-2
```

Próximas etapas

Depois de criar suas regras de cache de pull through, as próximas etapas são apresentadas a seguir:

- Crie um modelo de criação de repositório. Um modelo de criação de repositório oferece a você o controle para definir as configurações a serem usadas para novos repositórios criados pelo Amazon ECR em seu nome durante uma ação do cache de pull-through. Para ter mais

informações, consulte Modelos para controlar repositórios criados durante uma ação de extração do cache.

- Valide suas regras de cache de pull-through. Ao validar uma regra de cache de pull-through, o Amazon ECR faz uma conexão de rede com o registro upstream, verifica se ele pode acessar o segredo do Secrets Manager contendo as credenciais do registro upstream e se a autenticação foi bem-sucedida. Para ter mais informações, consulte Validando regras de pull through cache no Amazon ECR.
- Comece a usar suas regras de cache de pull-through. Para ter mais informações, consulte Extraindo uma imagem com uma regra de cache pull through no Amazon ECR.

Modelos para controlar repositórios criados durante uma ação de extração do cache

A funcionalidade de modelo de criação de repositório está em lançamento de visualização para o Amazon ECR e está sujeita a alterações. Durante essa pré-visualização pública, somente o AWS Management Console pode ser usado para gerenciar seus modelos de criação de repositórios.

Use os modelos de criação de repositórios do Amazon ECR para definir as configurações dos repositórios criados pelo Amazon ECR em seu nome durante uma ação de pull through cache. As configurações em um modelo de criação de repositório são aplicadas apenas durante a criação do repositório e não têm efeito sobre repositórios existentes ou repositórios criados usando qualquer outro método.

Os modelos de criação de repositórios não são compatíveis com as seguintes regiões:

- China (Pequim) (`cn-north-1`)
- China (Ningxia) (`cn-northwest-1`)
- AWS GovCloud (Leste dos EUA) (`us-gov-east-1`)
- AWS GovCloud (Oeste dos EUA) (`us-gov-west-1`)

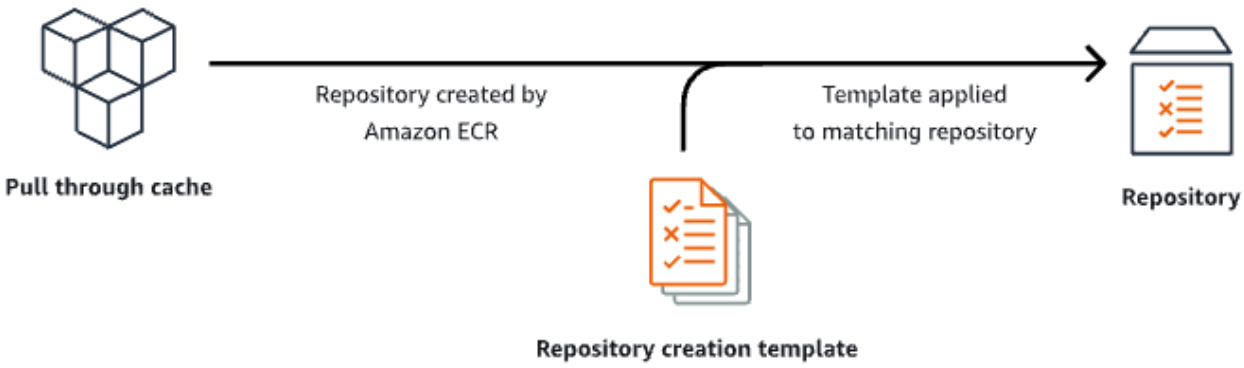
Como funcionam os modelos de criação de repositórios

Há momentos em que o Amazon ECR precisa criar um novo repositório privado em seu nome. Por exemplo, a primeira vez que você usa uma regra de cache de pull-through para recuperar o

conteúdo de um repositório upstream e armazená-lo em seu registro privado do Amazon ECR. Quando não há um modelo de criação de repositório que corresponda à sua regra de cache de pull-through, o Amazon ECR utiliza as configurações padrão para o novo repositório. Essas configurações padrão incluem desativar a imutabilidade da tag, usar criptografia `AES-256` e não aplicar nenhuma política de repositório ou ciclo de vida.

Usar um modelo de criação de repositório com um prefixo que corresponde a uma regra de cache de pull through oferece a capacidade de definir as configurações que o Amazon ECR aplica a novos repositórios criados por meio da ação de cache de pull-through. É possível definir a imutabilidade da tag, a configuração de criptografia, as permissões do repositório, a política de ciclo de vida e as tags de recursos para os novos repositórios.

O diagrama a seguir mostra o fluxo de trabalho que o Amazon ECR usa quando um modelo de criação de repositório é usado.



A seguir, descrevemos detalhadamente cada parâmetro no modelo de criação de repositório.

Prefixo

O prefixo é o prefixo do namespace do repositório a ser associado ao modelo. Todos os repositórios criados usando esse prefixo terão as configurações aplicadas que estão definidas nesse modelo. Por exemplo, um prefixo de se `prod` aplicaria a todos os repositórios começando com `prod/`. Por exemplo, um prefixo de `prod/team` se aplicaria a todos os repositórios começando com `prod/team/`.

Para aplicar um modelo a todos os repositórios em seu registro que não têm um modelo de criação associado, você pode usar `ROOT` como prefixo.

> **Important**
>
> Sempre há uma suposição / aplicada ao fim do prefixo. Se você especificar `ecr-public` como prefixo, o Amazon ECR tratará isso como `ecr-public/`. Ao usar uma regra de cache de pull-through, o prefixo do repositório que você especifica durante a criação da regra é o que você também deve especificar como prefixo do modelo de criação do repositório.

Descrição

Essa descrição do modelo é opcional e é usada para descrever a finalidade do modelo de criação do repositório.

Versão do modelo

A versão do modelo de criação de repositório a ser usada. Atualmente, apenas a versão `TV1` do modelo tem suporte.

Versão de configuração

A versão de configuração do repositório, o modelo a ser usado. Cada modelo deve incluir uma configuração de repositório. A versão padrão da configuração é `CV1` e consiste nas configurações de mutabilidade da tag de imagem, política de repositório e política de ciclo de vida.

Mutabilidade de tag de imagem

A configuração de mutabilidade da tag a ser usada para repositórios criados usando o modelo. Se este parâmetro for omitido, será usada a configuração padrão de MUTÁVEL, o que permitirá que as tags de imagem sejam substituídas. Essa é a configuração recomendada a ser usada para modelos usados em repositórios criados por ações de cache de pull-through. Isso garante que o Amazon ECR possa atualizar as imagens em cache quando as tags forem as mesmas.

Se IMUTÁVEL for especificada, todas as tags de imagem dentro do repositório serão imutáveis, o que impedirá que elas sejam substituídas.

Configuração de criptografia

A configuração de criptografia a ser usada para repositórios criados usando o modelo.

Se você usar o tipo de criptografia KMS, o conteúdo do repositório será criptografado usando criptografia do lado do servidor com uma chave AWS Key Management Service armazenada

em AWS KMS. Ao usar AWS KMS para criptografar seus dados, você pode usar a AWS KMS chave AWS gerenciada padrão para o Amazon ECR ou especificar sua própria AWS KMS chave, que você já criou. Para obter mais informações, consulte Proteção de dados usando criptografia do lado do servidor com uma AWS Key Management Service chave armazenada em AWS Key Management Service (SSE-KMS) no Guia do usuário do Amazon Simple Storage Service.

Se você usar o tipo de criptografia AES256, o Amazon ECR utilizará a criptografia do lado do servidor com chaves de criptografia gerenciadas pelo Amazon S3, que criptografam as imagens no repositório usando um algoritmo de criptografia AES-256. Para obter mais informações, consulte Proteção de dados usando criptografia no lado do servidor com chaves de criptografia gerenciadas pelo Amazon S3 (SSE-S3) no Manual do usuário do Amazon Simple Storage Service.

Permissões do repositório

A política de repositório a ser aplicada aos repositórios criados usando o modelo. Uma política de repositório utiliza permissões baseadas em recursos para controlar o acesso a um repositório. As permissões baseadas em recursos permitem especificar quais usuários ou funções do IAM têm acesso a um repositório e quais ações podem realizar nele. Por padrão, somente a AWS conta que criou o repositório tem acesso a um repositório. É possível aplicar um documento de política que concede ou nega permissões adicionais ao seu repositório. Para ter mais informações, consulte Políticas de repositório privado no Amazon ECR.

Política de ciclo de vida do repositório

A política de ciclo de vida a ser usada para repositórios criados usando o modelo. Uma política de ciclo de vida oferece mais controle sobre o gerenciamento do ciclo de vida das imagens em um repositório privado. Uma política de ciclo de vida é um conjunto de uma ou mais regras em que cada regra define uma ação do Amazon ECR. Isso permite a automação da limpeza de imagens de suas imagens de contêiner por imagens com validade prestes a expirar baseadas em idade ou número. Para ter mais informações, consulte Automatize a limpeza de imagens usando políticas de ciclo de vida no Amazon ECR.

Tags de recursos

As tags de recurso são metadados a serem aplicados ao repositório para ajudá-lo a categorizá-los e organizá-los. Cada tag consiste em uma chave e um valor opcional, ambos definidos por você.

Permissões do IAM para criar modelos de criação de repositórios

As permissões a seguir são necessárias para que uma entidade principal do IAM gerencie os modelos de criação de repositórios. Essas permissões devem ser concedidas usando uma política do IAM baseada em identidade.

- `ecr:CreateRepositoryCreationTemplate` - Concede permissão para criar o modelo de criação do repositório
- `ecr:DeleteRepositoryCreationTemplate` - Concede permissão para excluir o modelo de criação do repositório
- `ecr:PutLifecyclePolicy` - Concede permissão para criar uma política de ciclo de vida e aplicá-la a um repositório. Esta permissão é necessária apenas se o modelo de criação de repositório incluir uma política de ciclo de vida.
- `ecr:SetRepositoryPolicy` - Concede permissão para criar uma política de permissões para um repositório. Esta permissão é necessária apenas se o modelo de criação de repositório incluir uma política de repositório.
- `ecr:TagResource` - Concede permissão para adicionar tags de metadados a um recurso. Esta permissão é necessária apenas se o modelo de criação de repositório incluir tags de recursos.

Criação de um modelo de criação de repositório no Amazon ECR

É possível criar um modelo de criação de repositório para definir as configurações a serem usadas para repositórios criados pelo Amazon ECR em seu nome durante ações de cache de pull-through. Depois que o modelo de criação do repositório for criado, todos os novos repositórios criados durante as ações de extração do cache terão as configurações aplicadas. Isso não tem efeito em repositórios criados anteriormente.

Para criar um modelo de criação de repositório (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/
2. Na barra de navegação, escolha a região para criar o modelo de criação de repositório.
3. No painel de navegação, escolha Registro privado, Modelos de criação de repositório.
4. Na página Modelos de criação de repositório, escolha Criar modelo.
5. Na página Etapa 1: definir modelo, para Detalhes do modelo, escolha Um prefixo específico para aplicar o modelo a um prefixo de namespace de repositório específico ou escolha

Qualquer prefixo em seu registro ECR para aplicar o modelo a todos os repositórios que não correspondam a nenhum outro modelo na região.

a. Se você escolher Um prefixo específico, em Prefixo, especifique o prefixo do namespace do repositório ao qual aplicar o modelo. Sempre há uma suposição `/` aplicada ao fim do prefixo. Por exemplo, um prefixo de se `prod` aplicaria a todos os repositórios começando com `prod/`. Por exemplo, um prefixo de `prod/team` se aplicaria a todos os repositórios começando com `prod/team/`.

b. Se você escolher Qualquer prefixo em seu registro do ECR, o prefixo será definido como `ROOT`.

6. Em Descrição do modelo, especifique uma descrição opcional para o modelo e escolha Avançar.

7. Na página Etapa 2: adicionar configuração de criação de repositório, especifique a configuração de configuração do repositório a ser aplicada aos repositórios criados usando o modelo.

a. Para Mutabilidade de tag de imagem, escolha a configuração de mutabilidade de tags a ser usada. Para ter mais informações, consulte Impedindo que as tags de imagem sejam sobrescritas no Amazon ECR.

Quando a opção Mutável é selecionada, as tags de imagem podem ser sobrescritas. Essa é a configuração recomendada a ser usada para modelos usados em repositórios criados por ações de cache de pull-through. Isso garante que o Amazon ECR possa atualizar as imagens em cache quando as tags forem as mesmas.

Quando a opção Imutável é selecionada, as tags de imagem não podem ser sobrescritas. Após o repositório ser configurado para tags imutáveis, um erro `ImageTagAlreadyExistsException` é retornado se houver uma tentativa de enviar uma imagem com uma tag que já está no repositório. Quando a imutabilidade de tags está ativada para um repositório, isso afeta todas as tags e você não pode tornar algumas etiquetas imutáveis enquanto outras não.

b. Para Configuração de criptografia, escolha a configuração de criptografia a ser usada. Para ter mais informações, consulte Criptografia em repouso.

Quando AES-256 é selecionado, o Amazon ECR utiliza a criptografia do lado do servidor com chaves de criptografia gerenciadas pelo Amazon Simple Storage Service, o que criptografa seus dados em repouso usando o padrão de criptografia AES-256. Este é oferecido sem custo adicional.

Quando o AWS KMS é selecionado, o Amazon ECR usa criptografia do lado do servidor com chaves armazenadas em (). AWS Key Management Service AWS KMS Ao usar AWS KMS para criptografar seus dados, você pode usar a chave AWS gerenciada padrão, que é gerenciada pelo Amazon ECR, ou especificar sua própria AWS KMS chave, chamada de chave gerenciada pelo cliente.

> **Note**
>
> As configurações de criptografia para um repositório não podem ser alteradas após a criação do repositório.

c. Para permissões do repositório, especifique a política de permissões do repositório a ser aplicada aos repositórios criados usando esse modelo. Opcionalmente, você pode usar o menu suspenso para selecionar uma das amostras de JSON para os casos de uso mais comuns. Para ter mais informações, consulte Políticas de repositório privado no Amazon ECR.

d. Para a Política do ciclo de vida do repositório, especifique a política de ciclo de vida do repositório a ser aplicada aos repositórios criados usando esse modelo. Opcionalmente, você pode usar o menu suspenso para selecionar uma das amostras de JSON para os casos de uso mais comuns. Para ter mais informações, consulte Automatize a limpeza de imagens usando políticas de ciclo de vida no Amazon ECR.

e. Para AWS tags de repositório, especifique os metadados, na forma de pares de valores-chave, a serem associados aos repositórios criados usando esse modelo e escolha Avançar. Para ter mais informações, consulte Marcar um repositório privado no Amazon ECR.

8. Na página Etapa 3: revisar e criar, revise as configurações que você especificou para o modelo de criação do repositório. Escolha a opção Editar para fazer alterações. Escolha Criar quando você terminar.

Excluindo um modelo de criação de repositório no Amazon ECR

É possível excluir um modelo de criação de repositório se terminar de usá-lo. Depois que o modelo de criação do repositório for excluído, todos os repositórios criados durante uma ação de extração do cache terão as configurações padrão aplicadas.

Para excluir um modelo de criação de repositório ()AWS Management Console

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/
2. Na barra de navegação, escolha a região onde o modelo de criação de repositório que deseja excluir está localizado.
3. No painel de navegação, escolha Registro privado, Modelos de criação de repositório.
4. Na página Modelos de criação de repositório, selecione o modelo de criação de repositório a ser excluído.
5. No menu suspenso Ações, escolha Excluir.

Validando regras de pull through cache no Amazon ECR

Depois de criar uma regra de cache pull through, para registros upstream que exigem autenticação, você pode validar se a regra funciona corretamente. Ao validar uma regra de cache pull through, o Amazon ECR faz uma conexão de rede com o registro upstream, verifica se pode acessar o segredo do Secrets Manager contendo as credenciais do registro upstream e verifica se a autenticação foi bem-sucedida.

Antes de começar a trabalhar com suas regras de cache de pull through, verifique se você tem as permissões adequadas do IAM. Para ter mais informações, consulte Permissões do IAM necessárias para sincronizar um registro upstream com um registro privado do Amazon ECR.

Para validar uma regra de cache de pull-through (AWS Management Console)

Os seguintes passos mostram como validar uma regra de cache de pull-through usando o console do Amazon ECR.

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/
2. Na barra de navegação, selecione a Região que contém a regra do cache de pull-through a ser validada.
3. No painel de navegação, escolha Private registry (Registro privado), Pull through cache (Cache de pull-through).
4. Na página de Configuração do cache de pull-through, selecione a regra de pull through cache para validar. Em seguida, use o menu suspenso Ações e escolha Exibir detalhes.
5. Na página de detalhes da regra de cache de pull-through, use o menu suspenso Ações e escolha Verificar autenticação. O Amazon ECR exibirá um banner com o resultado.

6. Repita essas etapas para cada regra de cache de pull-through que deseja validar.

Para validar uma regra de cache de pull-through (AWS CLI)

O AWS CLI comando validate-pull-through-cache-rule é usado para validar uma regra de cache pull through para um registro privado do Amazon ECR. O exemplo a seguir usa o prefixo do `ecr-public` namespace. Substitua esse valor pelo valor do prefixo para validação da regra de cache de pull-through.

```
aws ecr validate-pull-through-cache-rule \
     --ecr-repository-prefix ecr-public \
     --region us-east-2
```

Na resposta, o parâmetro `isValid` indica se a validação foi bem-sucedida ou não. Se `true`, o Amazon ECR conseguisse acessar o registro upstream e a autenticação fosse bem-sucedida. Se `false` houve um problema e a validação falhou. O parâmetro `failure` indica a causa.

Extraindo uma imagem com uma regra de cache pull through no Amazon ECR

Os exemplos a seguir mostram a sintaxe do comando a ser usada ao extrair uma imagem usando uma regra de cache de pull- through. Se você receber um erro ao extrair uma imagem upstream usando uma regra de cache de pull-through, consulte Solução de problemas de recuperação de cache no Amazon ECR para ver os erros mais comuns e como resolvê-los.

Antes de começar a trabalhar com suas regras de cache de pull through, verifique se você tem as permissões adequadas do IAM. Para ter mais informações, consulte Permissões do IAM necessárias para sincronizar um registro upstream com um registro privado do Amazon ECR.

> **Note**
>
> Os exemplos a seguir usam os valores de namespace padrão do repositório Amazon ECR que eles usam. AWS Management Console Certifique-se de usar o URI do repositório privado do Amazon ECR que você configurou.

Para o Amazon ECR Public

```
docker pull aws_account_id.dkr.ecr.region.amazonaws.com/ecr-public/repository_name/
image_name:tag
```

Registro de contêineres Kubernetes

```
docker pull aws_account_id.dkr.ecr.region.amazonaws.com/kubernetes/repository_name/
image_name:tag
```

Quay

```
docker pull aws_account_id.dkr.ecr.region.amazonaws.com/quay/repository_name/
image_name:tag
```

Docker Hub

Para imagens oficiais do Docker Hub:

```
docker pull aws_account_id.dkr.ecr.region.amazonaws.com/docker-hub/
library/image_name:tag
```

> **Note**
>
> Para imagens oficiais do Docker Hub, o orefixo `/library` deve ser incluído. Para todos os outros repositórios do Docker Hub, você deve omitir o prefixo `/library`.

Para todas as outras imagens do Docker Hub:

```
docker pull aws_account_id.dkr.ecr.region.amazonaws.com/docker-hub/repository_name/
image_name:tag
```

GitHub Registro de contêiner

```
docker pull aws_account_id.dkr.ecr.region.amazonaws.com/github/repository_name/
image_name:tag
```

Microsoft Azure Container Registry

```
docker pull aws_account_id.dkr.ecr.region.amazonaws.com/azure/repository_name/
image_name:tag
```

GitLab Registro de contêiner

```
docker pull aws_account_id.dkr.ecr.region.amazonaws.com/gitlab/repository_name/
image_name:tag
```

Armazenando suas credenciais do repositório upstream em segredo AWS Secrets Manager

Ao criar uma regra de cache de pull-through para um repositório upstream que requer autenticação, você deve armazenar as credenciais em um segredo do Secrets Manager. Pode haver um custo para usar um segredo do Secrets Manager. Para obter mais informações, consulte Preços do AWS Secrets Manager.

Os procedimentos a seguir explicam como criar um segredo do Secrets Manager para cada repositório upstream compatível. Opcionalmente, você pode usar o fluxo de trabalho para criar regras de cache de pull-through no console do Amazon ECR para criar o segredo, em vez de criar o segredo usando o console do Secrets Manager. Para ter mais informações, consulte Criação de uma regra de cache pull through no Amazon ECR.

Docker Hub

Para criar um segredo do Secrets Manager para suas credenciais do Docker Hub (AWS Management Console)

1. Abra o console Secrets Manager em https://console.aws.amazon.com/secretsmanager/.
2. Escolha Armazenar Novo Segredo.
3. Na página Escolher o tipo de segredo, faça o seguinte:
   a. Em Secret type (Tipo de segredo), escolha Other type of secret (Outro tipo de segredo).
   b. Em pares de chave/valor, crie duas linhas para suas credenciais do Docker Hub. É possível armazenar até 65536 bytes no segredo.

i. Para o primeiro par chave/valor, especifique `username` como chave e seu nome de usuário do Docker Hub como valor.

ii. Para o segundo par chave/valor, especifique `accessToken` como chave e seu token de acesso ao Docker Hub como valor. Para obter mais informações sobre como criar um token de acesso do Docker Hub, consulte Criar e gerenciar tokens de acesso na documentação do Docker.

c. Em Chave de criptografia, mantenha o AWS KMS key valor padrão aws/secretsmanager e escolha Avançar. Não há custo para o uso dessa chave. Para obter mais informações, consulte Criptografia e decodificação secretas no Secrets Manager no Guia do usuário do AWS Secrets Manager .

> **Important**
>
> Você deve usar a chave de `aws/secretsmanager` criptografia padrão para criptografar seu segredo. O Amazon ECR não oferece suporte ao uso de uma chave gerenciada pelo cliente (CMK) para isso.

4. Na página Configurar segredo, faça o seguinte:

a. Insira um Secret name (Nome de segredo) descritivo e uma Description (Descrição). Os nomes de segredos devem conter de 1 a 512 caracteres Unicode e ter prefixo com `ecr-pullthroughcache/`.

> **Important**
>
> O Amazon ECR exibe AWS Management Console somente segredos do Secrets Manager com nomes usando o `ecr-pullthroughcache/` prefixo.

b. (Opcional) Na seção Tags (Etiquetas), adicione etiquetas ao segredo. Para estratégias de marcação, consulte os segredos do Tag Secrets Manager no Guia do usuário do AWS Secrets Manager . Não armazene informações sigilosas em etiquetas porque elas não são criptografadas.

c. (Opcional) Em Resource permissions (Permissões do recurso), para adicionar uma política de recursos ao segredo, escolha Edit permissions (Editar permissões). Para obter mais informações, consulte Anexo de uma política de permissões a um segredo do Secrets Manager no Guia do usuário do AWS Secrets Manager .

d. (Opcional) Em Replicar segredo, para replicar seu segredo para outro Região da AWS, escolha Replicar segredo. Você pode replicar seu segredo agora ou voltar e replicá-lo mais tarde. Para obter mais informações, consulte Replicar um segredo para outras Regiões no Guia do usuário do AWS Secrets Manager .

e. Selecione Next (Próximo).

5. (Opcional) Na página Configure rotation (Configurar alternância), habilite alternância automática para os segredos. Você também pode manter a alternância desabilitada por enquanto e habilitá-la mais tarde. Para obter mais informações, consulte o Guia do usuário do Secrets Manager no Guia do usuário do AWS Secrets Manager . Selecione Next (Próximo).

6. Na página Review (Revisar), revise os detalhes do segredo e escolha Store (Armazenar).

   O Secrets Manager retorna para a lista de segredos. Se o segredo não aparecer, escolha Refresh (Atualizar).

GitHub Container Registry

Para criar um segredo do Secrets Manager para suas credenciais do GitHub Container Registry ()AWS Management Console

1. Abra o console Secrets Manager em https://console.aws.amazon.com/secretsmanager/.
2. Escolha Armazenar Novo Segredo.
3. Na página Escolher o tipo de segredo, faça o seguinte:

   a. Em Secret type (Tipo de segredo), escolha Other type of secret (Outro tipo de segredo).

   b. Em pares de chave/valor, crie duas linhas para suas GitHub credenciais. É possível armazenar até 65536 bytes no segredo.

      i. Para o primeiro par chave/valor, especifique `username` como chave e seu GitHub nome de usuário como valor.

      ii. Para o segundo par chave/valor, especifique `accessToken` como chave e seu token de GitHub acesso como valor. Para obter mais informações sobre a criação de um token de GitHub acesso, consulte Gerenciando seus tokens de acesso pessoais na GitHub documentação.

   c. Em Chave de criptografia, mantenha o AWS KMS key valor padrão aws/secretsmanager e escolha Avançar. Não há custo para o uso dessa chave. Para obter mais informações,

consulte Criptografia e decodificação secretas no Secrets Manager no Guia do usuário do AWS Secrets Manager .

> **Important**
>
> Você deve usar a chave de `aws/secretsmanager` criptografia padrão para criptografar seu segredo. O Amazon ECR não oferece suporte ao uso de uma chave gerenciada pelo cliente (CMK) para isso.

4. Na página Configure secret (Configurar segredo), faça o seguinte:

   a. Insira um Secret name (Nome de segredo) descritivo e uma Description (Descrição). Os nomes de segredos devem conter de 1 a 512 caracteres Unicode e ter prefixo com `ecr-pullthroughcache/`.

   > **Important**
   >
   > O Amazon ECR exibe AWS Management Console somente segredos do Secrets Manager com nomes usando o `ecr-pullthroughcache/` prefixo.

   b. (Opcional) Na seção Tags (Etiquetas), adicione etiquetas ao segredo. Para estratégias de marcação, consulte os segredos do Tag Secrets Manager no Guia do usuário do AWS Secrets Manager . Não armazene informações sigilosas em etiquetas porque elas não são criptografadas.

   c. (Opcional) Em Resource permissions (Permissões do recurso), para adicionar uma política de recursos ao segredo, escolha Edit permissions (Editar permissões). Para obter mais informações, consulte Anexo de uma política de permissões a um segredo do Secrets Manager no Guia do usuário do AWS Secrets Manager .

   d. (Opcional) Em Replicar segredo, para replicar seu segredo para outro Região da AWS, escolha Replicar segredo. Você pode replicar seu segredo agora ou voltar e replicá-lo mais tarde. Para obter mais informações, consulte Replicar um segredo para outras Regiões no Guia do usuário do AWS Secrets Manager .

   e. Selecione Next (Próximo).

5. (Opcional) Na página Configure rotation (Configurar alternância), habilite alternância automática para os segredos. Você também pode manter a alternância desabilitada por enquanto e habilitá-la mais tarde. Para obter mais informações, consulte o Guia do usuário

do Secrets Manager no Guia do usuário do AWS Secrets Manager . Selecione Next (Próximo).

6. Na página Review (Revisar), revise os detalhes do segredo e escolha Store (Armazenar).

   O Secrets Manager retorna para a lista de segredos. Se o segredo não aparecer, escolha Refresh (Atualizar).

Microsoft Azure Container Registry

Para criar um segredo do Secrets Manager para suas credenciais do Microsoft Azure Container Registry (AWS Management Console)

1. Abra o console Secrets Manager em https://console.aws.amazon.com/secretsmanager/.
2. Escolha Armazenar Novo Segredo.
3. Na página Escolher o tipo de segredo, faça o seguinte:
   a. Em Secret type (Tipo de segredo), escolha Other type of secret (Outro tipo de segredo).
   b. Em pares de chave/valor, crie duas linhas para suas credenciais do GitHub. É possível armazenar até 65536 bytes no segredo.
      i. Para o primeiro par chave/valor, especifique `username` como chave e seu nome de usuário do Microsoft Azure Container Registry como valor.
      ii. Para o segundo par chave/valor, especifique `accessToken` como chave e seu nome de usuário do Microsoft Azure Container Registry como valor. Para obter mais informações sobre a criação de um token de acesso do Microsoft Azure, consulte Criar token - portal na documentação do Microsoft Azure.
   c. Em Chave de criptografia, mantenha o AWS KMS key valor padrão aws/secretsmanager e escolha Avançar. Não há custo para o uso dessa chave. Para obter mais informações, consulte Criptografia e decodificação secretas no Secrets Manager no Guia do usuário do AWS Secrets Manager .

> **Important**
>
> Você deve usar a chave de `aws/secretsmanager` criptografia padrão para criptografar seu segredo. O Amazon ECR não oferece suporte ao uso de uma chave gerenciada pelo cliente (CMK) para isso.

4. Na página Configure secret (Configurar segredo), faça o seguinte:

   a. Insira um Secret name (Nome de segredo) descritivo e uma Description (Descrição). Os nomes de segredos devem conter de 1 a 512 caracteres Unicode e ter prefixo com `ecr-pullthroughcache/`.

   > **Important**
   >
   > O Amazon ECR exibe AWS Management Console somente segredos do Secrets Manager com nomes usando o `ecr-pullthroughcache/` prefixo.

   b. (Opcional) Na seção Tags (Etiquetas), adicione etiquetas ao segredo. Para estratégias de marcação, consulte os segredos do Tag Secrets Manager no Guia do usuário do AWS Secrets Manager . Não armazene informações sigilosas em etiquetas porque elas não são criptografadas.

   c. (Opcional) Em Resource permissions (Permissões do recurso), para adicionar uma política de recursos ao segredo, escolha Edit permissions (Editar permissões). Para obter mais informações, consulte Anexo de uma política de permissões a um segredo do Secrets Manager no Guia do usuário do AWS Secrets Manager .

   d. (Opcional) Em Replicar segredo, para replicar seu segredo para outro Região da AWS, escolha Replicar segredo. Você pode replicar seu segredo agora ou voltar e replicá-lo mais tarde. Para obter mais informações, consulte Replicar um segredo para outras Regiões no Guia do usuário do AWS Secrets Manager .

   e. Selecione Next (Próximo).

5. (Opcional) Na página Configure rotation (Configurar alternância), habilite alternância automática para os segredos. Você também pode manter a alternância desabilitada por enquanto e habilitá-la mais tarde. Para obter mais informações, consulte o Guia do usuário do Secrets Manager no Guia do usuário do AWS Secrets Manager . Selecione Next (Próximo).

6. Na página Review (Revisar), revise os detalhes do segredo e escolha Store (Armazenar).

   O Secrets Manager retorna para a lista de segredos. Se o segredo não aparecer, escolha Refresh (Atualizar).

GitLab Container Registry

Para criar um segredo do Secrets Manager para suas credenciais do GitLab Container Registry ()AWS Management Console

1. Abra o console Secrets Manager em https://console.aws.amazon.com/secretsmanager/.
2. Escolha Armazenar Novo Segredo.
3. Na página Escolher o tipo de segredo, faça o seguinte:
   a. Em Secret type (Tipo de segredo), escolha Other type of secret (Outro tipo de segredo).
   b. Em pares de chave/valor, crie duas linhas para suas GitLab credenciais. É possível armazenar até 65536 bytes no segredo.
      i. Para o primeiro par chave/valor, especifique `username` como chave e seu nome de usuário do GitLab Container Registry como valor.
      ii. Para o segundo par chave/valor, especifique `accessToken` como chave e seu token de acesso do GitLab Container Registry como valor. Para obter mais informações sobre a criação de um token de acesso ao GitLab Container Registry, consulte Tokens de acesso pessoal, Tokens de acesso de grupo ou Tokens de acesso ao projeto, na GitLab documentação.
   c. Em Chave de criptografia, mantenha o AWS KMS key valor padrão aws/secretsmanager e escolha Avançar. Não há custo para o uso dessa chave. Para obter mais informações, consulte Criptografia e decodificação secretas no Secrets Manager no Guia do usuário do AWS Secrets Manager .

      > **Important**
      >
      > Você deve usar a chave de `aws/secretsmanager` criptografia padrão para criptografar seu segredo. O Amazon ECR não oferece suporte ao uso de uma chave gerenciada pelo cliente (CMK) para isso.

4. Na página Configure secret (Configurar segredo), faça o seguinte:
   a. Insira um Secret name (Nome de segredo) descritivo e uma Description (Descrição). Os nomes de segredos devem conter de 1 a 512 caracteres Unicode e ter prefixo com `ecr-pullthroughcache/`.

> **Important**
>
> O Amazon ECR exibe AWS Management Console somente segredos do Secrets Manager com nomes usando o `ecr-pullthroughcache/` prefixo.

b. (Opcional) Na seção Tags (Etiquetas), adicione etiquetas ao segredo. Para estratégias de marcação, consulte os segredos do Tag Secrets Manager no Guia do usuário do AWS Secrets Manager . Não armazene informações sigilosas em etiquetas porque elas não são criptografadas.

c. (Opcional) Em Resource permissions (Permissões do recurso), para adicionar uma política de recursos ao segredo, escolha Edit permissions (Editar permissões). Para obter mais informações, consulte Anexo de uma política de permissões a um segredo do Secrets Manager no Guia do usuário do AWS Secrets Manager .

d. (Opcional) Em Replicar segredo, para replicar seu segredo para outro Região da AWS, escolha Replicar segredo. Você pode replicar seu segredo agora ou voltar e replicá-lo mais tarde. Para obter mais informações, consulte Replicar um segredo para outras Regiões no Guia do usuário do AWS Secrets Manager .

e. Selecione Next (Próximo).

5. (Opcional) Na página Configure rotation (Configurar alternância), habilite alternância automática para os segredos. Você também pode manter a alternância desabilitada por enquanto e habilitá-la mais tarde. Para obter mais informações, consulte o Guia do usuário do Secrets Manager no Guia do usuário do AWS Secrets Manager . Selecione Next (Próximo).

6. Na página Review (Revisar), revise os detalhes do segredo e escolha Store (Armazenar).

   O Secrets Manager retorna para a lista de segredos. Se o segredo não aparecer, escolha Refresh (Atualizar).

Solução de problemas de recuperação de cache no Amazon ECR

Ao extrair uma imagem upstream usando uma regra de cache pull-through, os erros a seguir são os mais comuns que você pode receber.

O repositório não existe

Um erro que indica que o repositório não existe é mais frequentemente causado pela inexistência do repositório no seu registro privado do Amazon ECR ou porque a permissão `ecr:CreateRepository` não foi concedida à entidade principal do IAM que está extraindo a imagem upstream. Para resolver esse erro, você deve verificar se o URI do repositório no comando pull está correto, as permissões do IAM necessárias foram concedidas à entidade do IAM que está extraindo a imagem upstream ou se o repositório para a imagem upstream a ser enviada foi criado no registro privado do Amazon ECR antes da extração da imagem upstream. Para obter mais informações sobre as permissões necessárias do IAM, consulte Permissões do IAM necessárias para sincronizar um registro upstream com um registro privado do Amazon ECR

A seguir, temos um exemplo desse erro.

```
Error response from daemon: repository 111122223333.dkr.ecr.us-east-1.amazonaws.com/
ecr-public/amazonlinux/amazonlinux not found: name unknown: The repository with
 name 'ecr-public/amazonlinux/amazonlinux' does not exist in the registry with id
 '111122223333'
```

Imagem solicitada não encontrada

Um erro que indica que o repositório não pode ser encontrado é mais frequentemente causado pela inexistência do repositório upstream ou porque a permissão `ecr:BatchImportUpstreamImage` não foi concedida à entidade principal do IAM que está extraindo a imagem upstream, mas o repositório já está sendo criado no seu registro privado do Amazon ECR. Para resolver esse erro, você deve verificar se o nome da imagem upstream e da etiqueta da imagem está correto, se ela existe e se as permissões do IAM necessárias foram concedidas à entidade principal do IAM que está extraindo a imagem upstream. Para obter mais informações sobre as permissões necessárias do IAM, consulte Permissões do IAM necessárias para sincronizar um registro upstream com um registro privado do Amazon ECR.

A seguir, temos um exemplo desse erro.

```
Error response from daemon: manifest for 111122223333.dkr.ecr.us-
east-1.amazonaws.com/ecr-public/amazonlinux/amazonlinux:latest not found: manifest
 unknown: Requested image not found
```

403 Proibido ao extrair de um repositório do Docker Hub

Ao puxar de um repositório Docker Hub marcado como uma imagem oficial do Docker, você deve incluir o `/library/` no URI que você usa. Por exemplo, *aws_account_id*`.dkr.ecr.`*region*`.amazonaws.com/docker-hub/`**`library/`***image_name:tag*. Se você omitir as `/library/` imagens oficiais do Docker Hub, um erro `403 Forbidden` será retornado quando você tentar extrair a imagem usando uma regra de cache de pull-through. Para ter mais informações, consulte Extraindo uma imagem com uma regra de cache pull through no Amazon ECR.

A seguir, temos um exemplo desse erro.

```
Error response from daemon: failed to resolve reference "111122223333.dkr.ecr.us-
west-2.amazonaws.com/docker-hub/amazonlinux:2023": pulling from host
 111122223333.dkr.ecr.us-west-2.amazonaws.com failed with status code [manifests
 2023]: 403 Forbidden
```

Replicação de imagens privadas no Amazon ECR

É possível configurar o registro privado do Amazon ECR para oferecer suporte à replicação dos seus repositórios. O Amazon ECR oferece suporte à replicação entre regiões e entre contas. Para que ocorra replicação entre contas, a conta de destino deverá configurar uma política de permissões de registro para permitir que a replicação do registro de origem ocorra. Para ter mais informações, consulte Permissões de registro privado no Amazon ECR.

Tópicos

Considerações sobre a replicação de imagem privada

As seguintes informações devem ser consideradas ao usar replicação de imagem privada.

- Somente conteúdo do repositório enviado para um repositório após a replicação ser configurada é replicado. Nenhum conteúdo preexistente em um repositório é replicado. Depois que a replicação é configurada para um repositório, o Amazon ECR mantém o destino e a origem sincronizados.
- O nome do repositório permanecerá o mesmo em diferentes regiões e contas quando a replicação ocorrer. O Amazon ECR não suporta a alteração do nome do repositório durante a replicação.
- Na primeira vez que você configura seu registro privado para replicação, o Amazon ECR cria um perfil do IAM vinculada ao serviço em seu nome. A função do IAM vinculada ao serviço concede ao serviço de replicação do Amazon ECR a permissão necessária para criar repositórios e replicar imagens em seu registro. Para ter mais informações, consulte Uso de funções vinculadas ao serviço para o Amazon ECR.
- Para que a replicação entre contas ocorra, o destino do registro privado deve conceder permissão para permitir que o registro de origem replique suas imagens. Isso é feito definindo uma política de permissões de registro privado. Para ter mais informações, consulte Permissões de registro privado no Amazon ECR.
- Se a política de permissão para um registro privado for alterada para remover uma permissão, todas as replicações em andamento concedidas anteriormente poderão ser concluídas.
- Para que a replicação entre regiões ocorra, tanto a conta de origem quanto a conta de destino devem estar ativadas na região antes que qualquer ação de replicação ocorra dentro da região ou

tendo a região como destino. Para obter mais informações, consulte Como gerenciar regiões da AWS no Referência geral da Amazon Web Services.

- A replicação entre regiões não é suportada entre AWS partições. Por exemplo, um repositório em `us-west-2` não pode ser replicado para `cn-north-1`. Para obter mais informações sobre AWS partições, consulte o formato ARN AWS na Referência geral.
- A configuração de replicação para um registro privado pode conter até 25 destinos exclusivos em todas as regras, com um máximo de 10 regras no total. Cada regra pode conter até 100 filtros. Isso permite especificar regras separadas para repositórios contendo imagens usadas para produção e teste, por exemplo.
- A configuração de replicação oferece suporte à filtragem de quais repositórios em um registro privado são replicados especificando um prefixo de repositório. Para ver um exemplo, consulte Exemplo: configurar a replicação entre regiões usando um filtro de repositório.
- Uma ação de replicação ocorre apenas uma vez por envio de imagem. Por exemplo, se você configurou a replicação entre regiões do `us-west-2` para `us-east-1` e do `us-east-1` para `us-east-2`, uma imagem enviada para `us-west-2` replica somente para `us-east-1`, ela não é replicada novamente para `us-east-2`. Esse comportamento se aplica à replicação entre regiões e entre contas.
- A maioria das imagens replica-se em menos de 30 minutos, mas em casos raros a replicação pode demorar mais.
- Replicação do Registro não executa nenhuma ação de exclusão. Imagens e repositórios replicados podem ser excluídos manualmente quando não estão mais sendo usados.
- As políticas de repositório, incluindo políticas do IAM e políticas de ciclo de vida não são replicadas e não têm nenhum efeito além do repositório para o qual estão definidas.
- As configurações do repositório não são replicadas. As configurações de imutabilidade de etiqueta, varredura de imagem e criptografia KMS são desabilitadas por padrão em todos os repositórios criados devido a uma ação de replicação. A imutabilidade da etiqueta e a configuração de varredura de imagem podem ser alteradas após a criação do repositório. No entanto, a configuração só se aplica a imagens enviadas após a alteração da configuração.
- Se a imutabilidade da etiqueta estiver habilitada em um repositório e for replicada uma imagem que usa a mesma marca de uma imagem existente, a imagem será replicada, mas não conterá a etiqueta duplicada. Isso pode resultar na imagem ficar sem etiqueta.

Exemplos de replicação de imagens privadas para o Amazon ECR

Os exemplos a seguir mostram casos de uso comuns para replicação de imagens privadas. Se você configurar a replicação usando o AWS CLI, poderá usar os exemplos de JSON como ponto de partida ao criar seu arquivo JSON. Se você configurar a replicação usando o AWS Management Console, você verá um JSON semelhante ao revisar sua regra de replicação na página Revisar e enviar.

Exemplo: configurar a replicação entre regiões para uma única região de destino

A seguir, é mostrado um exemplo para configurar a replicação entre regiões em um único registro. Este exemplo pressupõe que o ID da conta seja `111122223333` e que você está especificando essa configuração de replicação em uma região diferente de `us-west-2`.

```
{
    "rules": [
        {
            "destinations": [
                {
                    "region": "us-west-2",
                    "registryId": "111122223333"
                }
            ]
        }
    ]
}
```

Exemplo: configurar a replicação entre regiões usando um filtro de repositório

O exemplo a seguir mostra um exemplo para configurar a replicação entre regiões para repositórios que correspondam a um valor de nome de prefixo. Este exemplo pressupõe que o ID da conta seja `111122223333` e que você está especificando essa configuração de replicação em uma região diferente de `us-west-1` e tem repositórios com prefixo `prod`.

```
{
 "rules": [{
```

```
  "destinations": [{
   "region": "us-west-1",
   "registryId": "111122223333"
  }],
  "repositoryFilters": [{
   "filter": "prod",
   "filterType": "PREFIX_MATCH"
  }]
 }]
}
```

Exemplo: configurar a replicação entre regiões para várias regiões de destino

A seguir, é mostrado um exemplo para configurar a replicação entre regiões em um único registro. Este exemplo pressupõe que o ID da conta seja `111122223333` e que você está especificando essa configuração de replicação em uma região diferente de `us-west-1` ou `us-west-2`.

```
{
    "rules": [
        {
            "destinations": [
                {
                    "region": "us-west-1",
                    "registryId": "111122223333"
                },
                {
                    "region": "us-west-2",
                    "registryId": "111122223333"
                }
            ]
        }
    ]
}
```

Exemplo: configurar replicação entre contas

A seguir, é mostrado um exemplo para configurar a replicação entre contas para o registo. Este exemplo configura a replicação para a conta `444455556666` e para a região `us-west-2`.

> **Important**
>
> Para que ocorra replicação entre contas, a conta de destino deve configurar uma política de permissões de registo para permitir que a replicação ocorra. Para obter mais informações, consulte Permissões de registro privado no Amazon ECR.

```
{
    "rules": [
        {
            "destinations": [
                {
                    "region": "us-west-2",
                    "registryId": "444455556666"
                }
            ]
        }
    ]
}
```

Exemplo: especificar várias regras em uma configuração

A seguir, é mostrado um exemplo para configurar várias regras de replicação para o seu registro. Este exemplo configura a replicação para a conta *111122223333* com uma regra que replica repositórios com um prefixo `prod` para a região `us-west-2` e repositórios com um prefixo `test` para a região `us-east-2`. Uma configuração de replicação pode conter até 10 regras, com cada regra especificando até 25 destinos.

```
{
 "rules": [{
   "destinations": [{
    "region": "us-west-2",
    "registryId": "111122223333"
   }],
   "repositoryFilters": [{
    "filter": "prod",
    "filterType": "PREFIX_MATCH"
   }]
  },
  {
```

```
    "destinations": [{
     "region": "us-east-2",
     "registryId": "111122223333"
    }],
    "repositoryFilters": [{
     "filter": "test",
     "filterType": "PREFIX_MATCH"
    }]
   }
  ]
}
```

Configurando a replicação de imagens privadas no Amazon ECR

Configure a replicação por região para seu registro privado. Você pode configurar a replicação entre regiões ou entre contas.

Para obter exemplos de como a replicação é comumente usada, consulte Exemplos de replicação de imagens privadas para o Amazon ECR.

Para configurar as definições de replicação do registo (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região para a qual definirá as configurações de replicação do registro.
3. No painel de navegação, escolha Private registry (Registro privado).
4. Na página Registro privado, na seção Replicação, selecione Editar.
5. Na página Replication (Replicação), selecione Add replication rule (Adicionar regra de replicação).
6. Na página Destination types (Tipos de destino), escolha se deseja ativar a replicação entre regiões, a replicação entre contas ou ambas e escolha Next (Próximo).
7. Se a replicação entre regiões estiver habilitada, então para Configure destination regions (Configurar regiões de destino), escolha um ou mais Destination regions (Regiões de destino) e, depois, escolha Next (Próximo).
8. Se a replicação entre contas estiver habilitada, então para Cross- accpint replication (Replicação entre contas), escolha a configuração de replicação entre contas para o registro. Para Destination account (Conta de destino), insira o ID da conta para a conta de destino e uma ou

mais Destination regions (Regiões de destino) para replicar. Selecione Destination account + (Conta de destino +) para configurar contas adicionais como destinos de replicação.

> **Important**
>
> Para que ocorra replicação entre contas, a conta de destino deve configurar uma política de permissões de registo para permitir que a replicação ocorra. Para obter mais informações, consulte Permissões de registro privado no Amazon ECR.

9. (Opcional) Na página Add filters (Adicionar filtros), especifique um ou mais filtros para a regra de replicação e escolhaAdd (Adicionar). Repita essa etapa para cada filtro que deseja associar à ação de replicação. Um filtro deve ser especificado como prefixo do nome do repositório. Se nenhum filtro for adicionado, o conteúdo de todos os repositórios será replicado. Selecione Next (Próximo) quando todos os filtros tiverem sido adicionados.
10. Na página Review and submit (Analisar e enviar), analise a configuração da regra de replicação e selecione Submit rule (Enviar regra).

Para configurar as definições de replicação do registo (AWS CLI)

1. Crie um arquivo JSON contendo as regras de replicação a serem definidas para o registro. Uma configuração de replicação pode conter até 10 regras, com até 25 destinos exclusivos entre todas as regras e 100 filtros por regra. Para configurar a replicação entre regiões em sua própria conta, especifique seu próprio ID de conta. Para obter mais exemplos, consulte Exemplos de replicação de imagens privadas para o Amazon ECR.

```
{
 "rules": [{
  "destinations": [{
   "region": "destination_region",
   "registryId": "destination_accountId"
  }],
  "repositoryFilters": [{
   "filter": "repository_prefix_name",
   "filterType": "PREFIX_MATCH"
  }]
 }]
}
```

2. Crie uma configuração de replicação para o seu registro.

```
aws ecr put-replication-configuration \
     --replication-configuration file://replication-settings.json \
     --region us-west-2
```

3. Confirme as configurações do seu registro.

```
aws ecr describe-registry \
     --region us-west-2
```

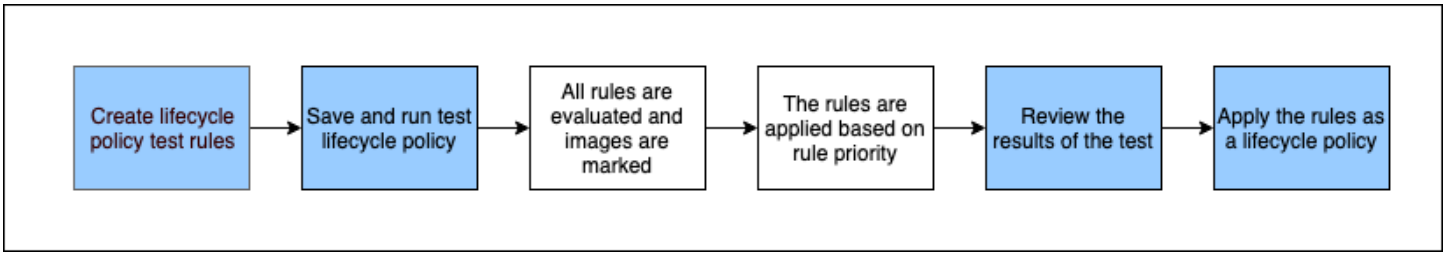
Automatize a limpeza de imagens usando políticas de ciclo de vida no Amazon ECR

As políticas de ciclo de vida do Amazon ECR fornecem mais controle sobre o gerenciamento de ciclo de vida de imagens em um repositório privado. Uma política de ciclo de vida contém uma ou mais regras, e cada regra define uma ação para o Amazon ECR. Com base nos critérios de expiração da política de ciclo de vida, as imagens expiram com base na idade ou na contagem em 24 horas. Quando o Amazon ECR executa uma ação com base em uma política de ciclo de vida, essa ação é capturada como um evento em. AWS CloudTrail Para ter mais informações, consulte Registrando ações do Amazon ECR com AWS CloudTrail.

Como funcionam as políticas de ciclo

Uma política de ciclo de vida consiste em uma ou mais regras que determinam quais imagens em um repositório devem perder a validade. Ao considerar o uso de políticas de ciclo de vida, é importante usar a visualização da política de ciclo de vida para confirmar de quais imagens a política de ciclo de vida expira a validade antes de aplicá-la a um repositório. Depois que uma política de ciclo de vida é aplicada a um repositório, você deve esperar que as imagens expirem dentro de 24 horas após atenderem aos critérios de expiração. Quando o Amazon ECR executa uma ação com base em uma política de ciclo de vida, ela é capturada como um evento no AWS CloudTrail. Para ter mais informações, consulte Registrando ações do Amazon ECR com AWS CloudTrail.

O diagrama a seguir mostra um fluxo de trabalho da política de ciclo de vida.



1. Crie uma ou mais regras de teste.
2. Salve as regras de teste e execute a visualização.
3. O avaliador de políticas de ciclo de vida percorre todas as regras e marca as imagens que cada regra afeta.
4. Em seguida, o avaliador de políticas de ciclo de vida aplica as regras, com base na prioridade da regra, e exibe quais imagens no repositório estão definidas para expirar a validade.

5. Revise os resultados do teste, certificando-se de que as imagens marcadas para expirar a validade são as que você pretendia.
6. Aplique as regras de teste como política de ciclo de vida para o repositório.
7. Depois que a política de ciclo de vida é criada, você deve esperar que as imagens expirem dentro de 24 horas após atenderem aos critérios de expiração.

Regras de avaliação de política de ciclo de vida

O avaliador da política de ciclo de vida é responsável por analisar o JSON de texto simples da política de ciclo de vida, avaliando todas as regras e aplicando essas regras com base na prioridade da regra às imagens no repositório. A seguir encontra-se a explicação mais detalhada da lógica do avaliador de políticas de ciclo de vida. Para ver exemplos, consulte Exemplos de políticas de ciclo de vida no Amazon ECR.

- Todas as regras são avaliadas ao mesmo tempo, independentemente da prioridade da regra. Depois que todas as regras são avaliadas, elas são aplicadas com base na prioridade da regra.
- Uma imagem é expirada por exatamente uma ou nenhuma regra.
- Uma imagem que corresponde aos requisitos de marcação de uma regra não pode ser expirada por uma regra com uma prioridade inferior.
- As regras nunca podem marcar imagens marcadas por regras de maior prioridade, mas ainda podem identificá-las como se não tivessem expirado.
- O conjunto de regras deve conter um conjunto exclusivo de prefixos de tags.
- Somente uma regra é permitida para selecionar imagens não marcadas.
- Se uma imagem for referenciada por uma lista de manifestos, ela não poderá expirar sem que a lista de manifestos seja excluída primeiro.
- A expiração é sempre solicitada por `pushed_at_time` e expira sempre as imagens mais antigas antes das mais novas.
- Uma regra de política de ciclo de vida pode especificar um `tagPatternList` ou `tagPrefixList`, mas não ambos. Porém, uma política de ciclo de vida pode conter várias regras em que regras diferentes usam listas de padrões e prefixos.
- Os parâmetros `tagPatternList` ou `tagPrefixList` apenas poderão ser usados se o `tagStatus` for `tagged`.
- Ao usar `tagPatternList`, uma imagem será correspondida com êxito se corresponder ao filtro de curinga. Por exemplo, se um filtro de `prod*` for aplicado, ele corresponderá aos repositórios

cujo nome começa com `prod`, como `prod`, `prod1` ou `production-team1`. Da mesma maneira, se um filtro de `*prod*` for aplicado, ele corresponderá aos repositórios cujo nome contenha `prod`, como `repo-production` ou `prod-team`.

> **Important**
>
> Existe um limite máximo de quatro curingas (*) por string. Por exemplo, `["*test*1*2*3", "test*1*2*3*"]` é válido, mas `["test*1*2*3*4*5*6"]` é inválido.

- Ao usar `tagPrefixList`, uma imagem será correspondida com êxito se todas as tags no valor `tagPrefixList` corresponderem a qualquer tag da imagem.
- O parâmetro `countUnit` só será usado se `countType` for `sinceImagePushed`.
- Com `countType = imageCountMoreThan`, as imagens são classificadas das mais novas para as mais antigas com base em `pushed_at_time` e, em seguida, todas as imagens acima da contagem especificada são expiradas.
- Com `countType = sinceImagePushed`, todas as imagens que tiverem o valor de `pushed_at_time` mais antigo do que o número de dias especificado com base em `countNumber` serão expiradas.

Criação de uma prévia da política de ciclo de vida no Amazon ECR

Você pode usar uma prévia da política de ciclo de vida para ver o impacto de uma política de ciclo de vida em um repositório de imagens antes de aplicá-la. É prática recomendada fazer uma visualização antes de aplicar uma política de ciclo de vida a um repositório.

> **Note**
>
> Se você estiver usando a replicação do Amazon ECR para fazer cópias de um repositório em diferentes regiões ou contas, observe que uma política de ciclo de vida só pode realizar uma ação em repositórios na região em que foi criada. Portanto, se você tiver a replicação ativada, convém criar uma política de ciclo de vida em cada região e conta para a qual estiver replicando os repositórios.

Para criar uma política de ciclo de vida (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.
2. Na barra de navegação, selecione a região que contém o repositório no qual a visualização de uma política de ciclo de vida será executada.
3. No painel de navegação, em Registro privado, escolha Repositórios.
4. Na página Repositórios privados, selecione um repositório e use o menu suspenso Ações para escolher Políticas de ciclo de vida.
5. Na página de regras de política de ciclo de vida do repositório, selecione Editar regras de teste, Criar regra.
6. Especifique os seguintes detalhes para cada regra da política de ciclo de vida de teste.

    a. Em Prioridade de regra, digite um número para a prioridade da regra. A prioridade da regra determina em que ordem as regras de políticas de ciclo de vida são aplicadas.

    b. Em Descrição da regra, digite uma descrição para a regra de política de ciclo de vida.

    c. Em Status da imagem, escolha Marcado (correspondência de curingas), Marcado (correspondência de prefixo), Sem etiqueta ou Qualquer.

    d. Se escolher Marcado (correspondência de curingas) para Status da imagem, em Especificar etiquetas para correspondência de curingas, você poderá especificar uma lista de etiquetas de imagem com um curinga (*) para agir com sua política de ciclo de vida. Por exemplo, se as suas imagens forem marcadas como `prod`, `prod1`, `prod2` e assim por diante, você especificaria `prod*` para agir em todas elas. Se você especificar várias tags, apenas imagens com todas as tags especificadas serão selecionadas.

    > **Important**
    >
    > Existe um limite máximo de quatro curingas (*) por string. Por exemplo, `["*test*1*2*3", "test*1*2*3*"]` é válido, mas `["test*1*2*3*4*5*6"]` é inválido.

    e. Se escolher Marcado (correspondência de prefixo) para Status da imagem e, em Especificar etiquetas para correspondência de prefixo, você poderá especificar uma lista de etiquetas de imagem nas quais agir com sua política de ciclo de vida.

    f. Em Critérios de correspondência, escolha Desde que a imagem foi enviada ou Contagem de imagens maior que e especifique um valor.

    g. Escolha Salvar.

7. Crie regras de política de ciclo de vida de teste adicionais repetindo as etapas de 5 a 7.
8. Para executar a visualização da política de ciclo de vida, escolha Salvar e executar teste.
9. Em Combinações de imagem para regras de ciclo de vida de teste, avalie o impacto da visualização da política de ciclo de vida.
10. Se você estiver satisfeito com os resultados da visualização, escolha Aplicar como política de ciclo de vida para criar uma política de ciclo de vida com as regras especificadas. Após aplicar uma política de ciclo de vida, você deve esperar que as imagens afetadas expirem em 24 horas.
11. Se não estiver satisfeito com os resultados da previsualização, poderá excluir uma ou mais regras de ciclo de vida de teste e criar uma ou mais regras para substituí-las e, em seguida, repetir o teste.

Criação de uma política de ciclo de vida para um repositório no Amazon ECR

Use uma política de ciclo de vida para criar um conjunto de regras que expiram imagens de repositório não utilizadas. Depois de criar uma política de ciclo de vida, as imagens afetadas expiram em 24 horas.

> **Note**
>
> Se você estiver usando a replicação do Amazon ECR para fazer cópias de um repositório em diferentes regiões ou contas, observe que uma política de ciclo de vida só pode realizar uma ação em repositórios na região em que foi criada. Portanto, se você tiver a replicação ativada, convém criar uma política de ciclo de vida em cada região e conta para a qual estiver replicando os repositórios.

Pré-requisito

Prática recomendada: crie uma prévia da política de ciclo de vida para verificar se as imagens expiradas de acordo com suas regras de política de ciclo de vida são o que você pretende. Para obter instruções, consulte Criação de uma prévia da política de ciclo de vida no Amazon ECR.

Como criar uma política de ciclo de vida (AWS Management Console)

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/repositories.

2. Na barra de navegação, selecione a região que contém o repositório para o qual uma política de ciclo de vida será criada.
3. No painel de navegação, em Registro privado, escolha Repositórios.
4. Na página Repositórios privados, selecione um repositório e use o menu suspenso Ações para escolher Políticas de ciclo de vida.
5. Na página de regras de política de ciclo de vida do repositório, selecione Criar regra.
6. Insira os seguintes detalhes para sua a regra da política de ciclo de vida.

    a. Em Prioridade de regra, digite um número para a prioridade da regra. A prioridade da regra determina em que ordem as regras de políticas de ciclo de vida são aplicadas.

    b. Em Descrição da regra, digite uma descrição para a regra de política de ciclo de vida.

    c. Em Status da imagem, escolha Marcado (correspondência de curingas), Marcado (correspondência de prefixo), Sem etiqueta ou Qualquer.

    d. Se escolher Marcado (correspondência de curingas) para Status da imagem, em Especificar etiquetas para correspondência de curingas, você poderá especificar uma lista de etiquetas de imagem com um curinga (*) para agir com sua política de ciclo de vida. Por exemplo, se as suas imagens forem marcadas como `prod`, `prod1`, `prod2` e assim por diante, você especificaria `prod*` para agir em todas elas. Se você especificar várias tags, apenas imagens com todas as tags especificadas serão selecionadas.

    > **Important**
    >
    > Existe um limite máximo de quatro curingas (*) por string. Por exemplo, `["*test*1*2*3", "test*1*2*3*"]` é válido, mas `["test*1*2*3*4*5*6"]` é inválido.

    e. Se escolher Marcado (correspondência de prefixo) para Status da imagem e, em Especificar etiquetas para correspondência de prefixo, você poderá especificar uma lista de etiquetas de imagem nas quais agir com sua política de ciclo de vida.

    f. Em Critérios de correspondência, escolha Desde que a imagem foi enviada ou Contagem de imagens maior que e especifique um valor.

    g. Escolha Salvar.

7. Crie regras de política de ciclo de vida adicionais repetindo as etapas de 5 a 7.

Como criar uma política de ciclo de vida (AWS CLI)

1. Obtenha o nome do repositório para o qual a política de ciclo de vida será criada.

```
aws ecr describe-repositories
```

2. Crie um arquivo local chamado `policy.json` com o conteúdo da política de ciclo de vida. Para ver exemplos de política do ciclo de vida, consulte Exemplos de políticas de ciclo de vida no Amazon ECR.
3. Crie uma política de ciclo de vida especificando o nome do repositório e referencie o arquivo JSON da política de ciclo de vida criado.

```
aws ecr put-lifecycle-policy \
      --repository-name repository-name \
      --lifecycle-policy-text file://policy.json
```

Exemplos de políticas de ciclo de vida no Amazon ECR

Veja a seguir exemplos de políticas de ciclo de vida que mostram a sintaxe.

Para ver mais informações sobre as propriedades da política, consultePropriedades da política de ciclo de vida no Amazon ECR. Para obter instruções sobre como criar uma política de ciclo de vida usando o AWS CLI, consulte. Como criar uma política de ciclo de vida (AWS CLI)

Modelo de política do ciclo de vida

O conteúdo da sua política de ciclo de vida é avaliado antes de ser associado a um repositório. Veja a seguir o modelo de sintaxe JSON de política de ciclo de vida.

```
{
        "rules": [
            {
                "rulePriority": integer,
                "description": "string",
                "selection": {
                    "tagStatus": "tagged"|"untagged"|"any",
                    "tagPatternList": list<string>,
                    "tagPrefixList": list<string>,
                    "countType": "imageCountMoreThan"|"sinceImagePushed",
                    "countUnit": "string",
```

```
                "countNumber": integer
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

Filtrar pela idade da imagem

O exemplo a seguir mostra a sintaxe da política de ciclo de vida de uma política que expira imagens com uma etiqueta que começa com `prod` usando um `tagPatternList` de `prod*` que também tenha mais de 14 dias.

```
{
    "rules": [
        {
            "rulePriority": 1,
            "description": "Expire images older than 14 days",
            "selection": {
                "tagStatus": "tagged",
                "tagPatternList": ["prod*"],
                "countType": "sinceImagePushed",
                "countUnit": "days",
                "countNumber": 14
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

Filtrar pela contagem da imagem

O exemplo a seguir mostra a sintaxe de política de ciclo de vida para uma política que mantém apenas uma imagem não marcada e expira todas as outras.

```
{
    "rules": [
        {
```

```
            "rulePriority": 1,
            "description": "Keep only one untagged image, expire all others",
            "selection": {
                "tagStatus": "untagged",
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

Filtrar por várias regras

Os exemplos a seguir usam várias regras em uma política de ciclo de vida. São fornecidos um repositório e uma política de ciclo de vida de exemplo com uma explicação do resultado.

Exemplo A

Conteúdo do repositório:

- Imagem A, Taglist: ["beta-1", "prod-1"], Enviada: 10 dias atrás
- Imagem B, Taglist: ["beta-2", "prod-2"], Enviada: 9 dias atrás
- Imagem C, Taglist: ["beta-3"], Enviada: 8 dias atrás

Texto da política de ciclo de vida:

```
{
    "rules": [
        {
            "rulePriority": 1,
            "description": "Rule 1",
            "selection": {
                "tagStatus": "tagged",
                "tagPatternList": ["prod*"],
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
```

```
                "type": "expire"
            }
        },
        {
            "rulePriority": 2,
            "description": "Rule 2",
            "selection": {
                "tagStatus": "tagged",
                "tagPatternList": ["beta*"],
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

A lógica dessa política de ciclo de vida seria:

- A regra 1 identifica as imagens marcadas com o prefixo `prod`. Ela deve marcar imagens, começando com a mais antiga, até que haja uma ou menos imagem restante correspondente. Ela marca a imagem A para expiração.
- A regra 2 identifica as imagens marcadas com o prefixo `beta`. Ela deve marcar imagens, começando com a mais antiga, até que haja uma ou menos imagem restante correspondente. Ela marca as imagens A e B para expiração. No entanto, a imagem A já foi vista pela Regra 1 e se a imagem B fosse expirada, ela violaria a Regra. Portanto, é ignorada.
- Resultado: a imagem A é expirada.

Exemplo B

Este é o mesmo repositório do exemplo anterior mas a solicitação de prioridade de regra é alterada para ilustrar o resultado.

Conteúdo do repositório:

- Imagem A, Taglist: ["beta-1", "prod-1"], Enviada: 10 dias atrás
- Imagem B, Taglist: ["beta-2", "prod-2"], Enviada: 9 dias atrás
- Imagem C, Taglist: ["beta-3"], Enviada: 8 dias atrás

Texto da política de ciclo de vida:

```
{
    "rules": [
        {
            "rulePriority": 1,
            "description": "Rule 1",
            "selection": {
                "tagStatus": "tagged",
                "tagPatternList": ["beta*"],
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        },
        {
            "rulePriority": 2,
            "description": "Rule 2",
            "selection": {
                "tagStatus": "tagged",
                "tagPatternList": ["prod*"],
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

A lógica dessa política de ciclo de vida seria:

- A regra 1 identifica as imagens marcadas com o prefixo `beta`. Ela deve marcar imagens, começando com a mais antiga, até que haja uma ou menos imagem restante correspondente. Ela vê todas as três imagens e marcaria as imagens A e B para expiração.
- A regra 2 identifica as imagens marcadas com o prefixo `prod`. Ela deve marcar imagens, começando com a mais antiga, até que haja uma ou menos imagem restante correspondente. A Regra 2 não veria nenhuma imagem porque todas as imagens disponíveis já teriam sido vistas pela Regra 1. Portanto, nenhuma imagem adicional seria marcada.

- Resultado: as imagens A e B são expiradas.

Filtrar por várias tags em uma única regra

Os seguintes exemplos especificam a sintaxe de política de ciclo de vida para vários padrões de etiqueta em uma única regra. São fornecidos um repositório e uma política de ciclo de vida de exemplo com uma explicação do resultado.

Exemplo A

Quando vários padrões de etiquetas são especificados em uma única regra, as imagens devem corresponder a todos os padrões de etiquetas listados.

Conteúdo do repositório:

- Imagem A, Taglist: ["alpha-1"], Enviada: 12 dias atrás
- Imagem B, Taglist: ["beta-1"], Enviada: 11 dias atrás
- Imagem C, Taglist: ["alpha-2", "beta-2"], Enviada: 10 dias atrás
- Imagem D, Taglist: ["alpha-3"], Enviada: 4 dias atrás
- Imagem E, Taglist: ["beta-3"], Enviada: 3 dias atrás
- Imagem F, Taglist: ["alpha-4", "beta-4"], Enviada: 2 dias atrás

```
{
    "rules": [
        {
            "rulePriority": 1,
            "description": "Rule 1",
            "selection": {
                "tagStatus": "tagged",
                "tagPatternList": ["alpha*", "beta*"],
                "countType": "sinceImagePushed",
                "countNumber": 5,
                "countUnit": "days"
            },
            "action": {
                "type": "expire"
            }
        }
```

```
    ]
}
```

A lógica dessa política de ciclo de vida seria:

- A regra 1 identifica as imagens marcadas com o prefixo `alpha` e `beta`. Ela vê as imagens C e F. A Regra 1 deve marcar imagens que têm mais de cinco dias, que seria a imagem C.
- Resultado: a imagem C é expirada.

Exemplo B

O exemplo a seguir ilustra as tags que não são exclusivas.

Conteúdo do repositório:

- Imagem A, Taglist: ["alpha-1", "beta-1", "gamma-1"], Enviada: 10 dias atrás
- Imagem B, Taglist: ["alpha-2", "beta-2"], Enviada: 9 dias atrás
- Imagem C, Taglist: ["alpha-3", "beta-3", "gamma-2"], Enviada: 8 dias atrás

```
{
    "rules": [
        {
            "rulePriority": 1,
            "description": "Rule 1",
            "selection": {
                "tagStatus": "tagged",
                "tagPatternList": ["alpha*", "beta*"],
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

A lógica dessa política de ciclo de vida seria:

- A regra 1 identifica as imagens marcadas com o prefixo `alpha` e `beta`. Ela vê todas as imagens. Ela deve marcar imagens, começando com a mais antiga, até que haja uma ou menos imagem restante correspondente. E marca as imagens A e B para expiração.
- Resultado: as imagens A e B são expiradas.

Filtrar todas as imagens

Os exemplos de política de ciclo de vida a seguir especificam todas as imagens com filtros diferentes. São fornecidos um repositório e uma política de ciclo de vida de exemplo com uma explicação do resultado.

Exemplo A

O exemplo a seguir mostra a sintaxe de política de ciclo de vida para uma política que é aplicada a todas as regras, mas mantém apenas uma imagem e expira todas as outras.

Conteúdo do repositório:

- Imagem A, Taglist: ["alpha-1"], Enviada: 4 dias atrás
- Imagem B, Taglist: ["beta-1"], Enviada: 3 dias atrás
- Imagem C, Taglist: [], Enviada: 2 dias atrás
- Imagem D, Taglist: ["alpha-2"], Enviada: 1 dia atrás

```
{
    "rules": [
        {
            "rulePriority": 1,
            "description": "Rule 1",
            "selection": {
                "tagStatus": "any",
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        }
    ]
```

```
}
```

A lógica dessa política de ciclo de vida seria:

- A regra 1 identifica todas as imagens. Ela visualiza as imagens A, B, C e. D. Deve expirar todas as imagens, exceto a mais recente. E marca as imagens A, B e C para expiração.
- Resultado: as imagens A, B e C são expiradas.

Exemplo B

O exemplo a seguir ilustra uma política de ciclo de vida que combina todos os tipos de regra em uma única política.

Conteúdo do repositório:

- Imagem A, Taglist: ["alpha-", "beta-1", "-1"], Enviada: 4 dias atrás
- Imagem B, Taglist: [], Enviada: 3 dias atrás
- Imagem C, Taglist: ["alpha-2"], Enviada: 2 dias atrás
- Imagem D, Taglist: ["git hash"], Enviada: 1 dia atrás
- Imagem E, Taglist: [], Enviada: 1 dia atrás

```
{
    "rules": [
        {
            "rulePriority": 1,
            "description": "Rule 1",
            "selection": {
                "tagStatus": "tagged",
                "tagPatternList": ["alpha"],
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        },
        {
            "rulePriority": 2,
            "description": "Rule 2",
```

```
            "selection": {
                "tagStatus": "untagged",
                "countType": "sinceImagePushed",
                "countUnit": "days",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        },
        {
            "rulePriority": 3,
            "description": "Rule 3",
            "selection": {
                "tagStatus": "any",
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

A lógica dessa política de ciclo de vida seria:

- A regra 1 identifica as imagens marcadas com o prefixo `alpha`. Ela identifica as imagens A e C. Deve manter a imagem mais recente e marcar o restante para expiração. Ela marca a imagem A para expiração.
- A regra 2 identifica as imagens não marcadas. Ela identifica as imagens B e E. Deve marcar todas as imagens com mais de um dia para expiração. E marca a imagem B para expiração.
- A regra 3 identifica todas as imagens. Ela identifica as imagens A, B, C, D e E. Deve manter a imagem mais recente e marcar o restante para expiração. No entanto, ela não pode marcar as imagens A, B, C ou E, pois elas foram identificadas por regras de maior prioridade. E marca a imagem D para expiração.
- Resultado: as imagens A, B e D são expiradas.

Propriedades da política de ciclo de vida no Amazon ECR

As políticas de ciclo de vida têm as seguintes propriedades.

Para ver exemplos de políticas de ciclo de vida, consulte. Exemplos de políticas de ciclo de vida no Amazon ECR Para obter instruções sobre como criar uma política de ciclo de vida usando o AWS CLI, consulte. Como criar uma política de ciclo de vida (AWS CLI)

Prioridade das regras

`rulePriority`

Tipo: inteiro

Obrigatório: sim

Define a ordem em que as regras são avaliadas, da menor para a maior. Uma regra de política de ciclo de vida com prioridade de 1 é aplicada primeiro, uma regra com prioridade de 2 é a próxima e assim por diante. Ao adicionar regras a uma política de ciclo de vida, você deve dar a elas um valor exclusivo para `rulePriority`. Os valores não precisam ser sequenciais entre as regras de uma política. Uma regra com um valor `tagStatus` de `any` deve ter o valor o mais alto para `rulePriority` e ser avaliada por último.

Descrição

`description`

Tipo: sequência

Obrigatório: não

(Opcional) Descreve a finalidade de uma regra em uma política de ciclo de vida.

Status da tag

`tagStatus`

Tipo: sequência

Obrigatório: sim

Determina se a regra da política de ciclo de vida que você está adicionando especifica uma tag para uma imagem. As opções aceitáveis são `tagged`, `untagged` ou `any`. Se você especificar `any`, todas as regras serão avaliadas segundo a regra Se você especificar `tagged`, você também deverá especificar um valor `tagPrefixList`. Se você especificar `untagged`, você deverá omitir `tagPrefixList`.

Lista de padrões de etiquetas

`tagPatternList`

Tipo: list[string]

Obrigatório: sim, se `tagStatus` estiver definido como marcado e se `tagPrefixList` não for especificado

Ao criar uma política de ciclo de vida para imagens marcadas, uma prática recomendada é usar um `tagPatternList` para especificar as etiquetas que expirarão. Você especifica uma lista separada por vírgulas de padrões de etiquetas de imagem que podem conter curingas (`*`) sobre os quais agir com sua política de ciclo de vida. Por exemplo, se suas imagens forem marcadas como `prod`, `prod1`, `prod2` e assim por diante, você deve usar a lista de padrões de etiquetas `prod*` para especificá-las. Se você especificar várias tags, apenas imagens com todas as tags especificadas serão selecionadas.

> **Important**
>
> Existe um limite máximo de quatro curingas (`*`) por string. Por exemplo, `["*test*1*2*3", "test*1*2*3*"]` é válido, mas `["test*1*2*3*4*5*6"]` é inválido.

Lista de prefixos de tags

`tagPrefixList`

Tipo: list[string]

Obrigatório: sim, se `tagStatus` estiver definido como marcado e se `tagPatternList` não for especificado

Usado somente se você tiver especificado `"tagStatus": "tagged"` e não estiver especificando `tagPatternList`. Você deve especificar uma lista separada por vírgulas de prefixos de tags de imagem na qual agir com política de ciclo de vida. Por exemplo, se suas imagens forem marcadas como `prod`, `prod1`, `prod2` e assim por diante, você deve usar o prefixo de tag `prod` para especificá-las. Se você especificar várias tags, apenas imagens com todas as tags especificadas serão selecionadas.

Tipo de contagem

`countType`

Tipo: sequência

Obrigatório: sim

Especifique um tipo de contagem a ser aplicado às imagens.

Se `countType` for definido como `imageCountMoreThan`, você também especificará `countNumber` para criar uma regra que define um limite no número de imagens que existem no repositório. Se `countType` for definido como `sinceImagePushed`, você também especificará `countUnit` e `countNumber` para especificar um limite de tempo nas imagens que existem no repositório.

Unidade de contagem

`countUnit`

Tipo: sequência

Exigido: sim, somente se `countType` for definido como `sinceImagePushed`

Especifique uma unidade de contagem de `days` para indicar como a unidade de tempo, além de `countNumber`, que é o número de dias.

Isso só deverá ser especificado quando `countType` for `sinceImagePushed`; um erro ocorrerá se você especificar uma unidade de contagem quando `countType` for qualquer outro valor.

Contagem numérica

`countNumber`

Tipo: inteiro

Obrigatório: sim

Especifique um número de contagem. Os valores aceitáveis são inteiros positivos (`0` não é um valor aceito).

Se o `countType` usado for `imageCountMoreThan`, o valor será o número máximo de imagens que você deseja manter no repositório. Se o `countType` usado for `sinceImagePushed`, o valor será o limite de idade máximo das imagens.

Ação

`type`

Tipo: sequência

Obrigatório: sim

Especifique um tipo de ação. O valor suportado é `expire`.

Segurança no Amazon Elastic Container Registry

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de uma arquitetura de data center e rede criada para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O modelo de responsabilidade compartilhada descreve isto como segurança da nuvem e segurança na nuvem.

- Segurança da nuvem — AWS é responsável por proteger a infraestrutura que executa AWS os serviços na AWS nuvem. AWS também fornece serviços que você pode usar com segurança. Auditores de terceiros testam e verificam regularmente a eficácia da nossa segurança como parte dos compliance programs AWS. Para saber mais sobre os programas de conformidade que se aplicam ao Amazon ECR, consulte Serviços da AWS no escopo pelo programa de conformidade.
- Segurança na nuvem — Sua responsabilidade é determinada pelo AWS serviço que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda a entender como aplicar o modelo de responsabilidade compartilhada ao usar o Amazon ECR. Os tópicos a seguir mostram como configurar o Amazon ECR para atender aos seus objetivos de segurança e conformidade. Você também aprende a usar outros AWS serviços que ajudam a monitorar e proteger seus recursos do Amazon ECR.

Tópicos

Gerenciamento de Identidade e Acesso para o Amazon Elastic Container Registry

AWS Identity and Access Management (IAM) é uma ferramenta AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. Os administradores

do IAM controlam quem pode ser autenticado (conectado) e autorizado (ter permissões) para usar recursos do Amazon ECR. O IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Tópicos

Público

A forma como você usa AWS Identity and Access Management (IAM) difere, dependendo do trabalho que você faz no Amazon ECR.

Usuário do serviço – Se você usar o serviço Amazon ECR para fazer sua tarefa, o administrador fornecerá as credenciais e as permissões de que você precisa. À medida que mais recursos do Amazon ECR forem usados para realizar o trabalho, talvez sejam necessárias permissões adicionais. Entender como o acesso é gerenciado pode ajudar você a solicitar as permissões corretas ao seu administrador. Se você não puder acessar um recurso no Amazon ECR, consulte Solução de problemas de identidade e acesso para o Amazon Elastic Container Registry.

Administrador do serviço – Se for o responsável pelos recursos do Amazon ECR em sua empresa, você provavelmente terá acesso total ao Amazon ECR. Cabe a você determinar quais funcionalidades e recursos do Amazon ECR os usuários do seu serviço devem 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 os Introdução ao IAM. Para saber mais sobre como a empresa pode usar o IAM com o Amazon ECR, consulte Como o Amazon Elastic Container Registry funciona com o IAM.

Administrador do IAM – Se você é um administrador do IAM, talvez queira saber detalhes sobre como pode escrever políticas para gerenciar o acesso ao Amazon ECR. Para ver exemplos das políticas baseadas em identidade do Amazon ECR que podem ser usadas no IAM, consulte Exemplos de políticas baseadas em identidade do Amazon Elastic Container Registry.

Autenticando com identidades

A autenticação é como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado (conectado AWS) como o Usuário raiz da conta da AWS, como usuário do IAM ou assumindo uma função do IAM.

Você pode entrar AWS como uma identidade federada usando credenciais fornecidas por meio de uma fonte de identidade. AWS IAM Identity Center Usuários (IAM Identity Center), a autenticação de login único da sua empresa e suas credenciais do Google ou do Facebook são exemplos de identidades federadas. Quando você faz login como identidade federada, o administrador já configurou anteriormente a federação de identidades usando perfis do IAM. Ao acessar AWS usando a federação, você está assumindo indiretamente uma função.

Dependendo do tipo de usuário que você é, você pode entrar no AWS Management Console ou no portal de AWS acesso. Para obter mais informações sobre como fazer login em AWS, consulte Como fazer login Conta da AWS no Guia do Início de Sessão da AWS usuário.

Se você acessar AWS programaticamente, AWS fornece um kit de desenvolvimento de software (SDK) e uma interface de linha de comando (CLI) para assinar criptograficamente suas solicitações usando suas credenciais. Se você não usa AWS ferramentas, você mesmo deve assinar as solicitações. Para obter mais informações sobre como usar o método recomendado para assinar solicitações por conta própria, consulte Assinatura de solicitações de AWS API no Guia do usuário do IAM.

Independente do método de autenticação usado, também pode ser exigido que você forneça informações adicionais de segurança. Por exemplo, AWS recomenda que você use a autenticação multifator (MFA) para aumentar a segurança da sua conta. Para saber mais, consulte Autenticação Multifator no AWS IAM Identity Center Guia do Usuário. Usar a autenticação multifator (MFA) na AWS no Guia do Usuário do IAM.

Conta da AWS usuário root

Ao criar uma Conta da AWS, você começa com uma identidade de login que tem acesso completo a todos Serviços da AWS os recursos da conta. Essa identidade é chamada de usuário Conta da AWS raiz e é acessada fazendo login com o endereço de e-mail e a senha que você usou para criar a

conta. É altamente recomendável não usar o usuário raiz para tarefas diárias. Proteja as credenciais do usuário raiz e use-as para executar as tarefas que somente ele pode 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.

Usuários e grupos do IAM

Um usuário do IAM é uma identidade dentro da sua Conta da AWS que tem permissões específicas para uma única pessoa ou aplicativo. Sempre que possível, recomendamos depender de 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 uso específicos que exijam credenciais de longo prazo com usuários do IAM, recomendamos alternar as chaves de acesso. Para obter mais informações, consulte Alterne 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 usar 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 nome de grupo IAMAdmins e atribuir a esse grupo permissões para administrar recursos do IAM.

Usuários são diferentes de perfis. Um usuário é exclusivamente associado a uma pessoa ou a um aplicativo, mas uma função pode ser assumida por qualquer pessoa que precisar dela. Os usuários têm credenciais permanentes de longo prazo, mas os perfis fornecem credenciais temporárias. Para saber mais, consulte Quando Criar um Usuário do IAM (Ao Invés de uma Função) no Guia do Usuário do IAM.

Perfis do IAM

Uma função do IAM é uma identidade dentro da sua Conta da AWS que tem permissões específicas. Ele é semelhante a um usuário do IAM, mas não está associado a uma pessoa específica. Você pode assumir temporariamente uma função do IAM no AWS Management Console trocando de funções. Você pode assumir uma função chamando uma operação de AWS API AWS CLI ou usando uma URL personalizada. Para obter mais informações sobre métodos para usar perfis, consulte Usando Funções do IAM no Guia do Usuário do IAM.

Funções 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 Criando um Perfil para um Provedor de Identidades Terceirizado no Guia do Usuário do IAM. Se você usa o IAM Identity Center, configure um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o IAM Identity Center correlaciona o conjunto de permissões a um perfil no IAM. Para obter informações sobre conjuntos de permissões, consulte Conjuntos de Permissões no AWS IAM Identity Center Manual do Usuário.

- Permissões de usuários temporárias do IAM: um usuário ou perfil do IAM pode assumir um perfil do IAM para obter temporariamente permissões diferentes para uma tarefa específica.
- Acesso entre contas: você pode usar um perfil do IAM para permitir que alguém (uma entidade principal confiável) acesse recursos na sua conta de uma conta diferente. As funções são a forma primária de conceder acesso entre contas. No entanto, com alguns Serviços da AWS, você pode anexar uma política diretamente a um recurso (em vez de usar uma função como proxy). Para aprender a diferença entre funções e políticas baseadas em recurso para acesso entre contas, consulte Como as Funções do IAM Diferem das Políticas Baseadas em Recurso no Guia do Usuário do IAM.
- Acesso entre serviços — Alguns Serviços da AWS usam recursos em outros Serviços da AWS. Por exemplo, quando você faz uma chamada em um serviço, é comum que esse serviço execute aplicativos no Amazon EC2 ou armazene objetos no Amazon S3. Um serviço pode fazer isso usando as permissões de chamada da entidade principal, uma função de serviço ou uma função vinculada ao serviço.
  - Sessões de acesso direto (FAS) — Quando você usa um usuário ou uma função do IAM para realizar ações AWS, você é considerado o principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O FAS usa as permissões do diretor chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) para fazer solicitações aos serviços posteriores. As solicitações do FAS são feitas somente quando um serviço recebe uma solicitação que requer interações com outros Serviços da AWS ou com recursos para ser concluída. 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.
  - Função de Serviço: uma função de serviço é uma função 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 Criando 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 ao serviço é um tipo de função de serviço vinculada a um. AWS service (Serviço da AWS) O serviço pode assumir o perfil de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não pode editar as permissões para funções vinculadas a serviço.

- Aplicativos em execução no Amazon EC2 — Você pode usar uma função do IAM para gerenciar credenciais temporárias para aplicativos que estão sendo executados em uma instância do EC2 e fazendo AWS CLI solicitações de API. AWS É preferível fazer isso a armazenar chaves de acesso na instância do EC2. Para atribuir uma AWS função a uma instância do EC2 e disponibilizá-la para todos os seus aplicativos, você cria um perfil de instância anexado à instância. 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 uma função do IAM para conceder permissões a aplicativos em execução nas instâncias do Amazon EC2 no Guia do usuário do IAM.

Para aprender se deseja usar perfis do IAM, consulte Quando Criar uma Função do IAM (em Vez de um Usuário) no Guia do Usuário do IAM.

Gerenciando acesso usando políticas

Você controla o acesso AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política é um objeto AWS que, quando associada a uma identidade ou recurso, define suas permissões. AWS avalia essas políticas quando um principal (usuário, usuário raiz ou sessão de função) faz uma solicitação. As permissões nas políticas determinam se a solicitação será permitida ou negada. A maioria das políticas é armazenada 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 políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

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 às funções e os usuários podem assumir as funções.

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 função da AWS Management Console AWS CLI, da 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 do IAM, grupo de usuários ou perfil do IAM. Essas políticas controlam quais ações os usuários e funções podem realizar, em quais recursos e em quais 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 também podem ser categorizadas como políticas em linha ou políticas gerenciadas. As políticas em linha são incorporadas diretamente a um único usuário, grupo ou função. As políticas gerenciadas são políticas autônomas que você pode associar a vários usuários, grupos e funções em seu Conta da AWS. As políticas AWS gerenciadas incluem políticas gerenciadas e políticas gerenciadas pelo cliente. Para saber como selecionar entre uma política gerenciada ou uma política em linha, consulte Selecionar entre políticas gerenciadas e políticas em linha no Guia do usuário do IAM.

Políticas baseadas em recursos

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de função do IAM e as políticas do bucket do Amazon S3. Em serviços que suportem políticas baseadas em recursos, 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 especificada pode executar nesse recurso e em que condições. Você deve especificar uma entidade principal em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Políticas baseadas em atributos são políticas em linha que estão localizadas nesse serviço. Você não pode usar políticas AWS gerenciadas do IAM em uma política baseada em recursos.

Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais 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 recurso 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 e dos seus limites de permissões. As políticas baseadas em atributo que especificam o usuário ou o 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 Entidades 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 (OU) em. AWS Organizations AWS Organizations é um serviço para agrupar e gerenciar centralmente vários Contas da AWS que sua empresa possui. 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 nas contas dos membros, incluindo cada uma Usuário raiz da conta da AWS. Para obter mais informações sobre o Organizações e SCPs, consulte Como os SCPs Funcionam no AWS Organizations Manual do Usuário do.
- Políticas de sessão: são políticas avançadas que você transmite como um parâmetro quando cria de forma programática uma sessão temporária para uma função 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 AWS determinar se uma solicitação deve ser permitida quando vários tipos de políticas estão envolvidos, consulte Lógica de avaliação de políticas no Guia do usuário do IAM.

Como o Amazon Elastic Container Registry funciona com o IAM

Antes de usar o IAM para gerenciar o acesso ao Amazon ECR, você deve entender quais recursos do IAM estão disponíveis para uso com o Amazon ECR. Para obter uma visão de alto nível de como o Amazon ECR e outros AWS serviços funcionam com o IAM, consulte AWS Serviços que funcionam com o IAM no Guia do usuário do IAM.

Tópicos

Políticas baseadas em identidade do Amazon ECR

Com as políticas baseadas em identidade do IAM, é possível especificar ações ou recursos permitidos ou negados, assim como as condições sob as quais as ações são permitidas ou negadas. O Amazon ECR oferece suporte a ações, chaves de condição e recursos específicos. Para conhecer todos os elementos usados em uma política JSON, consulte Referência de elementos de política JSON do IAM no Guia do usuário do IAM.

Ações

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

O elemento `Action` de uma política JSON descreve as ações que você pode usar para permitir ou negar acesso em uma política. As ações de política geralmente têm o mesmo nome da operação de AWS API associada. Existem algumas exceções, como ações somente de permissão, que não têm uma operação de API correspondente. Há também algumas operações que 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.

As ações de política no Amazon ECR usam o seguinte prefixo antes da ação: `ecr:`. Por exemplo, para conceder a alguém permissão para criar um repositório do Amazon ECR com a operação de API `CreateRepository` do Amazon ECR, inclua a ação `ecr:CreateRepository` na política da pessoa. As instruções de política devem incluir um elemento `Action` ou `NotAction`. O Amazon ECR define seu próprio conjunto de ações que descrevem as tarefas que você pode executar com esse serviço.

Para especificar várias ações em uma única instrução, separe-as com vírgulas, como segue:

```
"Action": [
```

```
      "ecr:action1",
      "ecr:action2"
```

Você também pode especificar várias ações usando caracteres curinga (*). Por exemplo, para especificar todas as ações que começam com a palavra `Describe`, inclua a seguinte ação:

```
"Action": "ecr:Describe*"
```

Para ver uma lista das ações do Amazon ECR, consulte Ações, recursos e chaves de condição do Amazon Elastic Container Registry no Manual do usuário do IAM.

Recursos

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

O elemento `Resource` de política 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 recurso usando seu nome do recurso 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 não compatíveis com permissões no nível de recurso, como operações de listagem, use um curinga (*) para indicar que a instrução se aplica a todos os recursos.

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

Um recurso do repositório do Amazon ECR tem o seguinte ARN:

```
arn:${Partition}:ecr:${Region}:${Account}:repository/${Repository-name}
```

Para obter mais informações sobre o formato dos ARNs, consulte Amazon Resource Names (ARNs) e AWS Service Namespaces.

Por exemplo, para especificar o repositório `my-repo` na região `us-east-1` em sua instrução, use o seguinte ARN:

```
"Resource": "arn:aws:ecr:us-east-1:123456789012:repository/my-repo"
```

Para especificar todos os repositórios que pertencem a uma conta específica, use o caractere curinga (*):

```
"Resource": "arn:aws:ecr:us-east-1:123456789012:repository/*"
```

Para especificar vários recursos em uma única instrução, separe os ARNs com vírgulas.

```
"Resource": [
      "resource1",
      "resource2"
```

Para ver uma lista dos tipos de recursos do Amazon ECR e seus ARNs, consulte Recursos definidos pelo Amazon Elastic Container Registry no Manual do usuário do IAM. Para saber com quais ações você pode especificar o ARN de cada recurso, consulte Ações definidas pelo Amazon Elastic Container Registry.

Chaves de condição

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

O elemento `Condition` (ou bloco `Condition`) permite especificar condições nas quais uma instrução estiver em vigor. O elemento `Condition` é opcional. Você pode criar expressões condicionais que usem operadores de condição, como “igual a” ou “menor que”, para corresponder a condição da política 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, AWS avalia a condição usando uma `OR` operação lógica. Todas as condições devem ser atendidas antes que as permissões da instrução sejam concedidas.

Você também pode usar variáveis de espaço reservado ao especificar 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.

AWS suporta chaves de condição globais e chaves de condição específicas do serviço. Para ver todas as chaves de condição AWS globais, consulte as chaves de contexto de condição AWS global no Guia do usuário do IAM.

O Amazon ECR define seu próprio conjunto de chaves de condição e também oferece suporte ao uso de algumas chaves de condição globais. Para ver todas as chaves de condição AWS globais, consulte Chaves de contexto de condição AWS global no Guia do usuário do IAM.

A maioria das ações do Amazon ECR oferecem suporte às chaves de condição `aws:ResourceTag` e `ecr:ResourceTag`. Para ter mais informações, consulte Usar controle de acesso baseado em tags.

Para ver uma lista das chaves de condição do Amazon ECR, consulte Chaves de condição definidas pelo Amazon Elastic Container Registry noManual do usuário do IAM. Para saber com quais ações e recursos você pode usar a chave de condição, consulte Ações definidas pelo Amazon Elastic Container Registry.

Exemplos

Para ver exemplos de políticas baseadas em identidade do Amazon ECR, consulte Exemplos de políticas baseadas em identidade do Amazon Elastic Container Registry.

Políticas baseadas em recurso do Amazon ECR

As políticas baseadas em recursos são documentos de políticas JSON que especificam quais ações uma entidade principal pode executar no recurso do Amazon ECR e em quais condições. O Amazon ECR oferece suporte a políticas de permissões baseadas em recursos para repositórios do Amazon ECR. As políticas baseadas em recursos permitem conceder permissão de uso a outras contas especificada por recurso. Também é possível usar uma política baseada em recurso para permitir que um serviço da AWS acesse seus repositórios do Amazon ECR.

Para permitir o acesso entre contas, é possível especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em recurso. Adicionar uma entidade principal entre contas à política baseada em atributo é apenas metade da tarefa de estabelecimento da relação de confiança. Quando o principal e o recurso estão em AWS contas diferentes, você também deve conceder permissão à entidade principal para acessar o recurso. Conceda permissão anexando 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 política baseada em identidade adicional será necessária. Para mais informações, consulte Como perfis do IAM diferem de políticas baseadas em recursos no Manual do usuário do IAM.

O serviço Amazon ECR oferece suporte somente a um tipo de política baseada em recurso, denominada política de repositório, que é anexada a um repositório. Essa política define quais entidades principais (contas, usuários, funções e usuários federados) podem realizar ações no repositório. Para saber como anexar uma política baseada em recurso a um repositório, consulte Políticas de repositório privado no Amazon ECR.

> **Note**
>
> Em uma política de repositório do Amazon ECR, o elemento de política `Sid` aceita caracteres e espaçamento adicionais que as políticas do IAM não aceitam.

Exemplos

Para ver exemplos de políticas baseadas em recursos do Amazon ECR, consulte Exemplos de políticas de repositório privado no Amazon ECR.

Autorização baseada em tags do Amazon ECR

É possível anexar tags a recursos do Amazon ECR ou informar tags em uma solicitação para o Amazon ECR. Para controlar o acesso baseado em tags, forneça informações sobre a tag no elemento de condição de uma política usando as chaves de condição `ecr:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* ou `aws:TagKeys`. Para obter mais informações sobre recursos de marcação do Amazon ECR, consulte Marcar um repositório privado no Amazon ECR.

Para visualizar um exemplo de política baseada em identidade para limitar o acesso a um recurso baseado em tags desse recurso, consulte Usar controle de acesso baseado em tags.

Perfis do IAM no Amazon ECR

Uma função do IAM é uma entidade dentro da sua AWS conta que tem permissões específicas.

Usar credenciais temporárias com o Amazon ECR

É possível usar credenciais temporárias para fazer login com federação, assumir um perfil do IAM ou assumir um perfil entre contas. Você obtém credenciais de segurança temporárias chamando operações de AWS STS API, como AssumeRoleou GetFederationToken.

O Amazon ECR suporta o uso de credenciais temporárias.

Funções vinculadas a serviço

As funções vinculadas ao serviço permitem que AWS os serviços acessem recursos em outros serviços para concluir uma ação em seu nome. Os perfis vinculados a serviço aparecem na sua conta do IAM e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não pode editar as permissões para perfis vinculados a serviço.

O Amazon ECR oferece suporte as funções vinculadas ao serviço. Para ter mais informações, consulte Uso de funções vinculadas ao serviço para o Amazon ECR.

Exemplos de políticas baseadas em identidade do Amazon Elastic Container Registry

Por padrão, usuários e funções não têm permissão para criar ou modificar recursos do Amazon ECR. Eles também não podem realizar tarefas usando a AWS API AWS Management Console, AWS Command Line Interface (AWS CLI) ou. Para conceder permissões de usuários para executar ações nos recursos que eles 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 Amazon ECR, inclusive o formato dos ARNs para cada um dos tipos de recurso, consulte Ações, recursos e chaves de condição do Amazon Elastic Container Registry na Referência de autorização do serviço.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documentos de política JSON, consulte Criar políticas na guia JSON no Guia do usuário do IAM.

Tópicos

- Melhores práticas de políticas
- Usar o console do Amazon ECR
- Permitir que os usuários visualizem suas próprias permissões
- Acesso a um repositório do Amazon ECR

Melhores práticas de políticas

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

- Comece com as políticas AWS gerenciadas e avance para as permissões de privilégios mínimos — Para começar a conceder permissões aos seus usuários e cargas de trabalho, use as políticas AWS gerenciadas que concedem permissões para muitos casos de uso comuns. Eles estão disponíveis no seu Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo AWS cliente que sejam específicas para seus casos de uso. Para obter mais informações, consulte Políticas Gerenciadas pela AWS ou AWS Políticas Gerenciadas para Funções de Trabalho no Guia do Usuário do IAM.
- Aplique permissões de privilégio mínimo: ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em 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.
- Utilize 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 recursos. Por exemplo, você pode gravar uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso às ações de serviço se elas forem usadas por meio de uma ação específica AWS service (Serviço da AWS), como AWS CloudFormation. Para obter mais informações, consulte Condição de Elementos de Política JSON do IAM no Guia do Usuário do IAM.
- Use o IAM Access Analyzer para validar suas políticas do IAM para garantir permissões seguras e funcionais: o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam o idioma de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de 100 verificações de política e ações recomendadas para ajudar você a criar políticas seguras e funcionais. Para obter mais informações, consulte Validação de Política do IAM Access Analyzer no Guia do Usuário do IAM.
- Exigir autenticação multifator (MFA) — Se você tiver um cenário que exija usuários do IAM ou um usuário root, ative Conta da AWS a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para obter

mais informações, consulte Configurando 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.

Usar o console do Amazon ECR

Para acessar o console da Amazon ECR, você deve ter um conjunto mínimo de permissões. Essas permissões devem permitir que você liste e visualize detalhes sobre os recursos do Amazon ECR em sua AWS conta. 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.

Para garantir que essas entidades ainda possam usar o console do Amazon ECR, adicione a política `AmazonEC2ContainerRegistryReadOnly` AWS gerenciada às entidades. Para obter mais informações, consulte Adição de permissões a um usuário no Manual do usuário do IAM:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ecr:GetAuthorizationToken",
                "ecr:BatchCheckLayerAvailability",
                "ecr:GetDownloadUrlForLayer",
                "ecr:GetRepositoryPolicy",
                "ecr:DescribeRepositories",
                "ecr:ListImages",
                "ecr:DescribeImages",
                "ecr:BatchGetImage",
                "ecr:GetLifecyclePolicy",
                "ecr:GetLifecyclePolicyPreview",
                "ecr:ListTagsForResource",
                "ecr:DescribeImageScanFindings"
            ],
            "Resource": "*"
        }
    ]
}
```

Você não precisa permitir permissões mínimas do console para usuários que estão fazendo chamadas somente para a API AWS CLI ou para a AWS API. Em vez disso, permita o acesso somente às ações que correspondem à operação da API que você está tentando executar.

Permitir que os usuários visualizem suas próprias permissões

Este exemplo mostra como criar uma política que permita que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou programaticamente usando a API AWS CLI ou AWS .

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

```
}
```

Acesso a um repositório do Amazon ECR

Neste exemplo, você deseja conceder a um usuário da sua AWS conta acesso a um dos seus repositórios do Amazon ECR,. `my-repo` Você também quer permitir que o usuário envie, extraia e liste imagens.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"ListImagesInRepository",
         "Effect":"Allow",
         "Action":[
            "ecr:ListImages"
         ],
         "Resource":"arn:aws:ecr:us-east-1:123456789012:repository/my-repo"
      },
      {
         "Sid":"GetAuthorizationToken",
         "Effect":"Allow",
         "Action":[
            "ecr:GetAuthorizationToken"
         ],
         "Resource":"*"
      },
      {
         "Sid":"ManageRepositoryContents",
         "Effect":"Allow",
         "Action":[
                "ecr:BatchCheckLayerAvailability",
                "ecr:GetDownloadUrlForLayer",
                "ecr:GetRepositoryPolicy",
                "ecr:DescribeRepositories",
                "ecr:ListImages",
                "ecr:DescribeImages",
                "ecr:BatchGetImage",
                "ecr:InitiateLayerUpload",
                "ecr:UploadLayerPart",
                "ecr:CompleteLayerUpload",
                "ecr:PutImage"
         ],
```

```
            "Resource":"arn:aws:ecr:us-east-1:123456789012:repository/my-repo"
        }
    ]
}
```

Usar controle de acesso baseado em tags

A ação da CreateRepository API do Amazon ECR permite que você especifique tags ao criar o repositório. Para ter mais informações, consulte Marcar um repositório privado no Amazon ECR.

Para permitir que os usuários marquem repositórios na criação, eles devem ter permissões para usar a ação que cria o recurso (por exemplo, `ecr:CreateRepository`). Se as tags forem especificadas na ação resource-creating, a Amazon executará autorização adicional na ação `ecr:CreateRepository` para verificar se os usuários têm permissões para criar tags.

É possível usar controle de acesso baseado em tags por meio de políticas do IAM. Veja os exemplos a seguir.

A política a seguir só permitiria que um usuário criasse ou marcasse um repositório como `key=environment,value=dev`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCreateTaggedRepository",
            "Effect": "Allow",
            "Action": [
                "ecr:CreateRepository"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/environment": "dev"
                }
            }
        },
        {
            "Sid": "AllowTagRepository",
            "Effect": "Allow",
            "Action": [
                "ecr:TagResource"
```

```
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/environment": "dev"
                }
            }
        }
    ]
}
```

A política a seguir concederia a um usuário acesso a todos os repositórios, a menos que eles estivessem marcados como `key=environment,value=prod`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ecr:*",
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": "ecr:*",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ecr:ResourceTag/environment": "prod"
                }
            }
        }
    ]
}
```

AWS políticas gerenciadas para o Amazon Elastic Container Registry

Uma política AWS gerenciada é uma política autônoma criada e administrada por AWS. AWS as políticas gerenciadas são projetadas para fornecer permissões para muitos casos de uso comuns, para que você possa começar a atribuir permissões a usuários, grupos e funções.

Lembre-se de que as políticas AWS gerenciadas podem não conceder permissões de privilégio mínimo para seus casos de uso específicos porque elas estão disponíveis para uso de todos os AWS clientes. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo cliente da específicas para seus casos de uso.

Você não pode alterar as permissões definidas nas políticas AWS gerenciadas. Se AWS atualizar as permissões definidas em uma política AWS gerenciada, a atualização afetará todas as identidades principais (usuários, grupos e funções) às quais a política está anexada. AWS é mais provável que atualize uma política AWS gerenciada quando uma nova AWS service (Serviço da AWS) é lançada ou novas operações de API são disponibilizadas para serviços existentes.

Para mais informações, consulte Políticas gerenciadas pela AWS no Manual do usuário do IAM.

O Amazon ECR fornece várias políticas gerenciadas que você pode anexar às identidades do IAM ou às instâncias do Amazon EC2. Essas políticas gerenciadas permitem diferentes níveis de controle sobre o acesso aos recursos do Amazon ECR e às operações de API. Para obter mais informações sobre cada operação de API mencionada nessas políticas, consulteAções na Referência da API do Amazon Elastic Container Registry.

Tópicos

- AmazonEC2ContainerRegistryFullAccess
- AmazonEC2ContainerRegistryPowerUser
- AmazonEC2ContainerRegistryReadOnly
- AWSECRPullThroughCache_ServiceRolePolicy
- ECRReplicationServiceRolePolicy
- Atualizações do Amazon ECR para políticas AWS gerenciadas

`AmazonEC2ContainerRegistryFullAccess`

É possível anexar a política `AmazonEC2ContainerRegistryFullAccess` a suas identidades do IAM.

Você pode usar essa política gerenciada como um ponto de partida para criar sua própria política do IAM com base em seus requisitos específicos. Por exemplo, você pode criar uma política especificamente para fornecer a um usuário ou a uma função acesso total de administrador para gerenciar o uso do Amazon ECR. O recurso Políticas de ciclo de vida do Amazon ECR permite que os clientes especifiquem o gerenciamento do ciclo de vida

das imagens em um repositório. Os eventos da política de ciclo de vida são relatados como CloudTrail eventos. O Amazon ECR está integrado AWS CloudTrail para que possa exibir seus eventos de política de ciclo de vida diretamente no console do Amazon ECR. A política gerenciada `AmazonEC2ContainerRegistryFullAccess` do IAM inclui a permissão `cloudtrail:LookupEvents` para facilitar esse comportamento.

Detalhes de permissão

Esta política inclui as seguintes permissões:

- `ecr` – Permite acesso total a todas as APIs do Amazon ECR.
- `cloudtrail`— Permite que os diretores pesquisem eventos de gerenciamento ou eventos do AWS CloudTrail Insights que são capturados por CloudTrail.

A política de `AmazonEC2ContainerRegistryFullAccess` é a seguinte.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ecr:*",
                "cloudtrail:LookupEvents"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:AWSServiceName": [
                        "replication.ecr.amazonaws.com"
                    ]
                }
            }
        }
    ]
```

```
}
```

AmazonEC2ContainerRegistryPowerUser

É possível anexar a política `AmazonEC2ContainerRegistryPowerUser` a suas identidades do IAM.

Essa política concede permissões administrativas que permitem que os usuários do IAM leiam e gravem nos repositórios, mas não permitem que eles excluam repositórios ou alterem os documentos de política aplicados a eles.

Detalhes de permissão

Esta política inclui as seguintes permissões:

- `ecr` – Permite que os principais leiam e gravem nos repositórios, e também que leiam as políticas de ciclo de vida. Os principais não recebem permissão para excluir repositórios ou alterar as políticas de ciclo de vida que são aplicadas a eles.

A política de `AmazonEC2ContainerRegistryPowerUser` é a seguinte.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ecr:GetAuthorizationToken",
                "ecr:BatchCheckLayerAvailability",
                "ecr:GetDownloadUrlForLayer",
                "ecr:GetRepositoryPolicy",
                "ecr:DescribeRepositories",
                "ecr:ListImages",
                "ecr:DescribeImages",
                "ecr:BatchGetImage",
                "ecr:GetLifecyclePolicy",
                "ecr:GetLifecyclePolicyPreview",
                "ecr:ListTagsForResource",
                "ecr:DescribeImageScanFindings",
                "ecr:InitiateLayerUpload",
                "ecr:UploadLayerPart",
                "ecr:CompleteLayerUpload",
```

```
                "ecr:PutImage"
            ],
            "Resource": "*"
        }
    ]
}
```

AmazonEC2ContainerRegistryReadOnly

É possível anexar a política `AmazonEC2ContainerRegistryReadOnly` a suas identidades do IAM.

Esta política concede permissões de acesso somente para leitura ao Amazon ECR. Isso inclui a capacidade de listar repositórios e imagens dentro dos repositórios. Inclui também a capacidade de extrair imagens do Amazon ECR com a CLI do Docker.

Detalhes de permissão

Esta política inclui as seguintes permissões:

- `ecr` – permite que os principais leiam repositórios e suas respectivas políticas de ciclo de vida.

A política de `AmazonEC2ContainerRegistryReadOnly` é a seguinte.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ecr:GetAuthorizationToken",
                "ecr:BatchCheckLayerAvailability",
                "ecr:GetDownloadUrlForLayer",
                "ecr:GetRepositoryPolicy",
                "ecr:DescribeRepositories",
                "ecr:ListImages",
                "ecr:DescribeImages",
                "ecr:BatchGetImage",
                "ecr:GetLifecyclePolicy",
                "ecr:GetLifecyclePolicyPreview",
                "ecr:ListTagsForResource",
                "ecr:DescribeImageScanFindings"
```

```
            ],
            "Resource": "*"
        }
    ]
}
```

AWSECRPullThroughCache_ServiceRolePolicy

Não é possível anexar a política do IAM gerenciada `AWSECRPullThroughCache_ServiceRolePolicy` às suas entidades do IAM. Essa política é anexada a uma função vinculada a serviço que permite que o Amazon ECR envie imagens para seus repositórios por meio do fluxo de trabalho do cache de pull-through. Para ter mais informações, consulte Função vinculada ao serviço do Amazon ECR para cache de pull-through.

ECRReplicationServiceRolePolicy

Não é possível anexar a política do IAM gerenciada `ECRReplicationServiceRolePolicy` às suas entidades do IAM. Esta política é anexada a uma função vinculada ao serviço que permite ao Amazon ECR realizar ações em seu nome. Para ter mais informações, consulte Uso de funções vinculadas ao serviço para o Amazon ECR.

Atualizações do Amazon ECR para políticas AWS gerenciadas

Veja detalhes sobre as atualizações das políticas AWS gerenciadas do Amazon ECR desde o momento em que esse serviço começou a rastrear essas alterações. Para obter alertas automáticos sobre alterações feitas nesta página, inscreva-se no feed RSS na página Histórico de documentos do Amazon ECR.

Alteração	Descrição	Data
AWSECRPullThroughCache_ServiceRolePolicy: atualizar para uma política existente	O Amazon ECR adicionou novas permissões à política `AWSECRPullThroughCache_ServiceRolePolicy` . Essas permissões permitem que o Amazon ECR recupere o conteúdo criptografado de um segredo	15 de novembro de 2023

Alteração	Descrição	Data
	do Secrets Manager. Isso é necessário ao usar uma regra de cache de pull-through para armazenar em cache imagens de um registro upstream que requer autenticação.	
AWSECRPullThroughCache_ServiceRolePolicy – Nova política	O Amazon ECR adicionou uma nova política. Essa política está associada à função vinculada ao serviço `AWSServiceRoleForECRPullThroughCache` para o recurso de cache de pull-through.	29 de novembro de 2021
ECR ReplicationServiceRolePolicy — Nova política	O Amazon ECR adicionou uma nova política. Essa política está associada à função vinculada ao serviço `AWSServiceRoleForECRReplication` para o recurso de replicação.	4 de dezembro de 2020
AmazonEC2 Container Registry FullAccess — Atualização de uma política existente	O Amazon ECR adicionou novas permissões à política `AmazonEC2ContainerRegistryFullAccess`. Essas permissões permitem que os principais criem a função vinculada ao serviço do Amazon ECR.	4 de dezembro de 2020

Alteração	Descrição	Data
AmazonEC2 Container Registry ReadOnly — Atualização de uma política existente	O Amazon ECR adicionou novas permissões à política `AmazonEC2ContainerRegistryReadOnly` que permite que os principais leiam políticas de ciclo de vida, listem tags e descrevam as descobertas de digitalização para as imagens.	10 de dezembro de 2019
AmazonEC2 Container Registry PowerUser — Atualização de uma política existente	O Amazon ECR adicionou novas permissões à política `AmazonEC2ContainerRegistryPowerUser`. Elas permitem que os principais leiam políticas de ciclo de vida, listem tags e descrevam as descobertas de digitalização para as imagens.	10 de dezembro de 2019
AmazonEC2 Container Registry FullAccess — Atualização de uma política existente	O Amazon ECR adicionou novas permissões à política `AmazonEC2ContainerRegistryFullAccess`. Eles permitem que os diretores consultem eventos de gerenciamento ou eventos do AWS CloudTrail Insights que são capturados por CloudTrail.	10 de novembro de 2017

Alteração	Descrição	Data
AmazonEC2 Container Registry ReadOnly — Atualização de uma política existente	O Amazon ECR adicionou novas permissões à política `AmazonEC2ContainerRegistryReadOnly`. Elas permitem que os principais descrevam imagens do Amazon ECR.	11 de outubro de 2016
AmazonEC2 Container Registry PowerUser — Atualização de uma política existente	O Amazon ECR adicionou novas permissões à política `AmazonEC2ContainerRegistryPowerUser`. Elas permitem que os principais descrevam imagens do Amazon ECR.	11 de outubro de 2016
Amazon EC2 Container Registry ReadOnly — Nova política	O Amazon ECR adicionou uma nova política que concede permissões somente de leitura para o Amazon ECR. Essas permissões incluem a capacidade de listar repositórios e imagens nos repositórios. Incluem também a capacidade de extrair imagens do Amazon ECR com a CLI do Docker.	21 de dezembro de 2015

Alteração	Descrição	Data
Amazon EC2 Container Registry PowerUser — Nova política	O Amazon ECR adicionou uma nova política que concede permissões administrativas que permitem que os usuários leiam e gravem nos repositórios, mas não permitem que eles excluam repositórios ou alterem os documentos de política aplicados a eles.	21 de dezembro de 2015
Amazon EC2 Container Registry FullAccess — Nova política	O Amazon ECR adicionou uma nova política. Essa política concede ao acesso total ao Amazon ECR.	21 de dezembro de 2015
O Amazon ECR passou a monitorar alterações	O Amazon ECR começou a monitorar as alterações nas políticas AWS gerenciadas.	24 de junho de 2021

Uso de funções vinculadas ao serviço para o Amazon ECR

O Amazon Elastic Container Registry (Amazon ECR) AWS Identity and Access Management usa funções vinculadas ao serviço (IAM) para fornecer as permissões necessárias para usar a replicação e aproveitar os recursos de cache. A função vinculada ao serviço é um tipo especial de função do IAM vinculada diretamente ao Amazon ECR. A função vinculada ao serviço é predefinida pelo Amazon ECR. Ele inclui todas as permissões que o serviço requer para oferecer suporte à replicação e recursos de cache para o seu registro privado. Após você configurar a replicação ou o cache de pull-through para o registro, uma função vinculada ao serviço é criada automaticamente em seu nome. Para ter mais informações, consulte Configurações de registro privado no Amazon ECR.

Uma função vinculada ao serviço facilita a configuração da replicação e do cache de pull-through com o Amazon ECR. Isso porque, assim, você não precisa adicionar manualmente todas as permissões necessárias. O Amazon ECR define as permissões de suas funções vinculadas ao serviço e, a não ser que definido de outra forma, somente o Amazon ECR pode assumir suas

funções. As permissões definidas incluem a política de confiança e a política de permissões. A política de permissões não pode ser anexada a nenhuma outra entidade do IAM.

Você só poderá excluir a função vinculada ao serviço após desabilitar a replicação ou o cache de pull-through no seu registro. Isso garante que você não remova inadvertidamente as permissões que o Amazon ECR requer para esses recursos.

Para obter informações sobre outros produtos que oferecem suporte a funções vinculadas a serviços, consulte Serviços da AWS que funcionam com o IAM. Nessa página para vinculação, procure os serviços que têm Yes (Sim) na coluna Service-linked role (Função vinculada ao serviço). Escolha um Yes (Sim) com um link para ver a documentação da função vinculada a serviço para esse serviço.

Tópicos

- Regiões suportadas para a funções vinculadas a serviço do Amazon ECR
- Função vinculada ao serviço do Amazon ECR para replicação
- Função vinculada ao serviço do Amazon ECR para cache de pull-through

Regiões suportadas para a funções vinculadas a serviço do Amazon ECR

O Amazon ECR oferece suporte a funções vinculadas a serviços em todas as regiões em que o serviço do Amazon ECR está disponível. Para obter mais informações sobre a disponibilidade regional do Amazon ECR, consulte Regiões e endpoints da AWS.

Função vinculada ao serviço do Amazon ECR para replicação

O Amazon ECR usa uma função vinculada ao serviço chamada AWSServiceRoleForECRReplicationque permite que o Amazon ECR replique imagens em várias contas.

Permissões de função vinculada ao serviço para o Amazon ECR

A função AWSServiceRoleForECRReplication vinculada ao serviço confia nos seguintes serviços para assumir a função:

- `replication.ecr.amazonaws.com`

A política de permissões da função do `ECRReplicationServiceRolePolicy` a seguir permite que o Amazon ECR use as seguintes ações em todos os recursos:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ecr:CreateRepository",
                "ecr:ReplicateImage"
            ],
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> `ReplicateImage` é uma API interna que o Amazon ECR usa para replicação e não pode ser chamada diretamente.

Você deve configurar permissões para que uma entidade do IAM (por exemplo, um usuário, grupo ou função) crie, edite ou exclua uma função vinculada ao serviço. Para obter mais informações, consulte Permissões de função vinculada ao serviço no Manual do usuário do IAM.

Criar uma função vinculada ao serviço para o Amazon ECR

Você não precisa criar manualmente a função vinculada ao serviço Amazon ECR. Quando você define as configurações de replicação do seu registro na AWS Management Console, na ou na AWS API AWS CLI, o Amazon ECR cria a função vinculada ao serviço para você.

Se você excluir essa função vinculada ao serviço e precisar criá-la novamente, poderá usar esse mesmo processo para recriar a função em sua conta. Quando você define as configurações de replicação para o registro, o Amazon ECR cria a função vinculada ao serviço para você novamente.

Editar uma função vinculada ao serviço do Amazon ECR

O Amazon ECR não permite a edição manual da função vinculada ao AWSServiceRoleForECRReplication serviço. Depois que você criar um perfil vinculado ao serviço, não poderá alterar o nome do perfil, pois várias entidades podem fazer referência ao perfil. No entanto, você pode editar a descrição da função usando o IAM. Para ter mais informações, consulte Editar um perfil vinculado ao serviço no Guia do usuário do IAM.

Exclusão de função vinculada ao serviço para o Amazon ECR

Se você não precisar mais usar um recurso ou serviço que requer um perfil vinculado ao serviço, é recomendável excluí-lo. Dessa forma, você não tem uma entidade não utilizada que não seja monitorada ou mantida ativamente. Contudo, você deve remover a configuração de replicação do seu registro em todas as regiões para poder excluir manualmente a função vinculada ao serviço.

> **Note**
>
> Se você tentar excluir recursos enquanto o serviço Amazon ECR ainda está usando as funções, sua ação de exclusão pode falhar. Se isso acontecer, aguarde alguns minutos e tente novamente.

Para excluir os recursos do Amazon ECR usados pelo AWSServiceRoleForECRReplication

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/
2. Na barra de navegação, selecione a região na qual a configuração de replicação está definida.
3. No painel de navegação, escolha Private registry (Registro privado).
4. Na página Private registry (Registro privado), na seção Replicação configuration (Configuração de replicação), selecione Edit (Editar).
5. Para excluir todas as suas regras de replicação, escolha Delete all (Excluir tudo). Essa etapa requer confirmação.

Como excluir manualmente a função vinculada a serviço usando o IAM

Use o console do IAM AWS CLI, o ou a AWS API para excluir a função AWSServiceRoleForECRReplicationvinculada ao serviço. Para mais informações, consulte Excluir um perfil vinculado ao serviço no Guia do usuário do IAM.

Função vinculada ao serviço do Amazon ECR para cache de pull-through

O Amazon ECR usa uma função vinculada ao serviço chamada AWSServiceRoleForECRPullThroughCacheque dá permissão para o Amazon ECR realizar ações em seu nome para concluir ações de pull through cache. Para obter mais informações sobre o cache de pull-through, consulte Sincronize um registro upstream com um registro privado do Amazon ECR.

Permissões de função vinculada ao serviço para o Amazon ECR

A função AWSServiceRoleForECRPullThroughCachevinculada ao serviço confia no serviço a seguir para assumir a função.

- `pullthroughcache.ecr.amazonaws.com`

Detalhes da permissão

A política de permissões da `AWSECRPullThroughCache_ServiceRolePolicy` está vinculada à função vinculada ao serviço. Essa política gerenciada concede permissão ao Amazon ECR para realizar as seguintes ações. Para ter mais informações, consulte `AWSECRPullThroughCache_ServiceRolePolicy`.

- `ecr` - Permite que o serviço Amazon ECR envie imagens para um repositório privado.
- `secretsmanager:GetSecretValue`— Permite que o serviço Amazon ECR recupere o conteúdo criptografado de um AWS Secrets Manager segredo. Isso é necessário ao usar uma regra de cache de pull-through para armazenar em cache imagens de um registro upstream que requer autenticação em seu registro privado. Essa permissão se aplica apenas a segredos com o prefixo de nome `ecr-pullthroughcache/`.

A política da `AWSECRPullThroughCache_ServiceRolePolicy` contém os elementos a seguir:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ECR",
            "Effect": "Allow",
            "Action": [
                "ecr:GetAuthorizationToken",
                "ecr:BatchCheckLayerAvailability",
                "ecr:InitiateLayerUpload",
                "ecr:UploadLayerPart",
                "ecr:CompleteLayerUpload",
                "ecr:PutImage"
            ],
            "Resource": "*"
        },
        {
```

```
            "Sid": "SecretsManager",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": "arn:aws:secretsmanager:*:*:secret:ecr-pullthroughcache/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceAccount": "${aws:PrincipalAccount}"
                }
            }
        }
    ]
}
```

Você deve configurar permissões para que uma entidade do IAM (por exemplo, um usuário, grupo ou função) crie, edite ou exclua uma função vinculada ao serviço. Para obter mais informações, consulte Permissões de perfil vinculado a serviços no Guia do usuário do IAM.

Criar uma função vinculada ao serviço para o Amazon ECR

Você não precisa criar manualmente a função vinculada ao serviço do Amazon ECR para cache de pull-through. Quando você cria uma regra de cache pull through para seu registro privado na AWS Management Console, na ou na AWS API AWS CLI, o Amazon ECR cria a função vinculada ao serviço para você.

Se você excluir essa função vinculada ao serviço e precisar criá-la novamente, poderá usar esse mesmo processo para recriar a função em sua conta. Quando você cria uma regra de cache de pull-through para seu registro privado, o Amazon ECR cria a função vinculada ao serviço para você novamente, caso ela ainda não exista.

Editar uma função vinculada ao serviço do Amazon ECR

O Amazon ECR não permite a edição manual da função vinculada ao AWSServiceRoleForECRPullThroughCacheserviço. Após a criação da função vinculada ao serviço, você não poderá alterar o nome da função, pois várias entidades podem fazer referência à função. No entanto, você pode editar a descrição da função usando o IAM. Para ter mais informações, consulte Editar um perfil vinculado ao serviço no Guia do usuário do IAM.

Exclusão de função vinculada ao serviço para o Amazon ECR

Se você não precisar mais usar um recurso ou serviço que requer um perfil vinculado ao serviço, é recomendável excluí-lo. Dessa forma, você não tem uma entidade não utilizada que não seja monitorada ou mantida ativamente. Contudo, você deve remover as regras de cache de pull-through do seu registro em todas as regiões para poder excluir manualmente a função vinculada ao serviço.

> **Note**
>
> Se você tentar excluir recursos enquanto o serviço Amazon ECR ainda estiver usando as funções, sua ação de exclusão poderá falhar. Se isso acontecer, aguarde alguns minutos e tente novamente.

Para excluir os recursos do Amazon ECR usados pela função vinculada ao serviço AWSServiceRoleForECRPullThroughCache

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/
2. Na barra de navegação, selecione a região na qual suas regras de cache de pull-through são criadas.
3. No painel de navegação, escolha Private registry (Registro privado).
4. Na página Private registry (Registro privado), na seção Pull through cache configuration (Configuração de cache de pull-through), escolha Edit (Editar).
5. Para cada regra de cache de pull-through que você criou, selecione a regra e escolha Delete rule (Excluir regra).

Como excluir manualmente a função vinculada a serviço usando o IAM

Use o console do IAM AWS CLI, o ou a AWS API para excluir a função AWSServiceRoleForECRPullThroughCachevinculada ao serviço. Para mais informações, consulte Excluir um perfil vinculado ao serviço no Guia do usuário do IAM.

Solução de problemas de identidade e acesso para o Amazon Elastic Container Registry

Use as informações a seguir para ajudar a diagnosticar e corrigir problemas comuns que você possa encontrar ao trabalhar com a Amazon ECR e o IAM.

Tópicos

Não tenho autorização para executar uma ação na Amazon ECR

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 `ecr:`*`GetWidget`* fictícias.

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

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

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

Não estou autorizado a realizar o meu pedido: PassRole

Se receber uma mensagem de erro informando que você não tem autorização para executar a ação `iam:PassRole`, suas políticas devem ser atualizadas para permitir a transmissão de um perfil ao Amazon ECR.

Alguns Serviços da AWS permitem que você passe uma função existente para esse serviço em vez de criar uma nova função de serviço ou uma função vinculada ao serviço. Para fazer isso, é preciso ter permissões para passar o perfil para o serviço.

O erro exemplificado a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta usar o console para executar uma ação na Amazon ECR. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

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

Quero permitir que pessoas de fora da minha Conta da AWS acessem meus recursos do Amazon ECR

Você pode criar uma função 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 recursos ou listas de controle de acesso (ACLs), você pode usar políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se o Amazon ECR suporta a esses recursos, consulte Como o Amazon Elastic Container Registry funciona com o IAM.
- Para saber como fornecer acesso aos seus recursos em todos os Contas da AWS que você possui, consulte Como fornecer acesso a um usuário do IAM em outro Conta da AWS que você possui no Guia do usuário do IAM.
- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte Como fornecer acesso Contas da AWS a terceiros no Guia do usuário do IAM.
- Saiba como conceder acesso por meio da federação de identidades consultando Concedendo Acesso a Usuários Autenticados Externamente (Federação de Identidades) no Guia do Usuário do IAM.
- Para saber a diferença entre usar funções e políticas baseadas em recursos para acesso entre contas, consulte Como as funções do IAM diferem de políticas baseadas em recursos no Manual do usuário do IAM.

Proteção de dados no Amazon ECR

O modelo de responsabilidade AWS compartilhada O modelo se aplica à proteção de dados no Amazon Elastic Container Service. Conforme descrito neste modelo, AWS é responsável por

proteger a infraestrutura global que executa todos os Nuvem AWS. Você é responsável por manter o controle sobre seu conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos Serviços da AWS que usa. Para ter mais informações sobre a privacidade de dados, consulte as Perguntas frequentes sobre privacidade de dados. Para ter mais informações sobre a proteção de dados na Europa, consulte a AWS postagem do blog Shared Responsibility Model and GDPR no AWS Blog de segurança da.

Para fins de proteção de dados, recomendamos que você proteja Conta da AWS as credenciais e configure usuários individuais com AWS IAM Identity Center ou AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com os recursos. AWS Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure a API e o registro de atividades do usuário com AWS CloudTrail.
- Use soluções de AWS criptografia, juntamente com todos os controles de segurança padrão Serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sigilosos armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-2 ao acessar AWS por meio de uma interface de linha de comando ou de uma API, use um endpoint 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 email dos seus clientes, em marcações ou campos de formato livre, como um campo Name (Nome). Isso inclui quando você trabalha com o Amazon ECS ou outros Serviços da AWS usando o console, a API ou os AWS SDKs. AWS CLI 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 você fornecer um URL para um servidor externo, recomendemos fortemente que não sejam incluídas informações de credenciais no URL para validar a solicitação a esse servidor.

Tópicos

- Criptografia em repouso

Criptografia em repouso

O Amazon ECR armazena imagens em buckets do Amazon S3 gerenciados pelo Amazon ECR. Por padrão, o Amazon ECR usa criptografia no lado do servidor com chaves de criptografia gerenciadas pelo Amazon S3 que criptografa seus dados em repouso usando um algoritmo de criptografia AES-256. Isso não requer nenhuma ação da sua parte e é oferecido sem custo adicional. Para obter mais informações, consulte Proteção de dados usando criptografia no lado do servidor com chaves de criptografia gerenciadas pelo Amazon S3 (SSE-S3) no Manual do usuário do Amazon Simple Storage Service.

Para obter mais controle sobre a criptografia dos seus repositórios Amazon ECR, você pode usar a criptografia do lado do servidor com chaves KMS armazenadas em (). AWS Key Management Service AWS KMS Ao usar AWS KMS para criptografar seus dados, você pode usar o padrão Chave gerenciada pela AWS, que é gerenciado pelo Amazon ECR, ou especificar sua própria chave KMS (chamada de chave gerenciada pelo cliente). Para obter mais informações, consulte Proteção de dados usando criptografia do lado do servidor com chaves KMS armazenadas em AWS KMS (SSE-KMS) no Guia do usuário do Amazon Simple Storage Service.

Cada repositório do Amazon ECR tem uma configuração de criptografia, que é definida quando o repositório é criado. Você pode usar configurações de criptografia diferentes em cada repositório. Para ter mais informações, consulte Criação de um repositório privado do Amazon ECR para armazenar imagens.

Quando um repositório é criado com a AWS KMS criptografia ativada, uma chave KMS é usada para criptografar o conteúdo do repositório. Além disso, o Amazon ECR adiciona uma AWS KMS concessão à chave KMS com o repositório Amazon ECR como principal beneficiário.

A próxima seção fornece uma compreensão de alto nível de como o Amazon ECR é integrado com o AWS KMS para criptografar e descriptografar seus repositórios:

1. Ao criar um repositório, o Amazon ECR envia uma DescribeKeychamada para AWS KMS validar e recuperar o Amazon Resource Name (ARN) da chave KMS especificada na configuração de criptografia.
2. O Amazon ECR envia duas CreateGrantsolicitações AWS KMS para criar concessões na chave KMS para permitir que o Amazon ECR criptografe e descriptografe dados usando a chave de dados.
3. Ao enviar uma imagem, é feita uma solicitação de GenerateDatachave AWS KMS que especifica a chave KMS a ser usada para criptografar a camada e o manifesto da imagem.

4. AWS KMS gera uma nova chave de dados, a criptografa sob a chave KMS especificada e envia a chave de dados criptografada para ser armazenada com os metadados da camada de imagem e o manifesto da imagem.
5. Ao extrair uma imagem, é feita uma solicitação de Decrypt AWS KMS, especificando a chave de dados criptografada.
6. AWS KMS descriptografa a chave de dados criptografada e envia a chave de dados descriptografada para o Amazon S3.
7. A chave de dados é usada para descriptografar a camada de imagem antes que a camada de imagem seja extraída.
8. Quando um repositório é excluído, o Amazon ECR envia duas RetireGrantsolicitações AWS KMS para retirar as concessões criadas para o repositório.

Considerações

Os seguintes pontos devem ser considerados ao usar a AWS KMS criptografia com o Amazon ECR.

- Se você criar seu repositório Amazon ECR com criptografia KMS e não especificar uma chave KMS, o Amazon ECR usa um Chave gerenciada pela AWS com o alias por padrão. `aws/ecr` Essa chave KMS é criada em sua conta na primeira vez que você cria um repositório com criptografia KMS habilitada.
- Quando você usa a criptografia KMS com sua própria chave KMS, a chave deve existir na mesma região que seu repositório.
- As concessões que o Amazon ECR cria em seu nome não devem ser revogadas. Se você revogar a concessão que dá permissão ao Amazon ECR para usar as AWS KMS chaves em sua conta, o Amazon ECR não poderá acessar esses dados, criptografar novas imagens enviadas ao repositório ou descriptografá-las quando forem retiradas. Quando você revoga uma concessão do Amazon ECR, a alteração ocorre imediatamente. Para revogar direitos de acesso, você deve excluir o repositório em vez de revogar a concessão. Quando um repositório é excluído, o Amazon ECR retira as concessões em seu nome.
- Há um custo associado ao uso de AWS KMS chaves. Para obter mais informações, consulte Preços do AWS Key Management Service.

Permissões obrigatórias do IAM

Ao criar ou excluir um repositório do Amazon ECR com criptografia no lado do servidor usando o AWS KMS, as permissões necessárias dependem da chave KMS específica que você está usando.

Permissões do IAM necessárias ao usar o Chave gerenciada pela AWS for Amazon ECR

Por padrão, quando a AWS KMS criptografia está habilitada para um repositório Amazon ECR, mas nenhuma chave KMS é especificada, a para Chave gerenciada pela AWS Amazon ECR é usada. Quando a chave KMS AWS gerenciada do Amazon ECR é usada para criptografar um repositório, qualquer diretor que tenha permissão para criar um repositório também pode ativar a criptografia no repositório. AWS KMS No entanto, o principal do IAM que exclui o repositório deve ter a permissão `kms:RetireGrant`. Isso permite a retirada das concessões que foram adicionadas à AWS KMS chave quando o repositório foi criado.

O exemplo a seguir da política do IAM pode ser adicionado como uma política em linha a um usuário para garantir que ele tenha as permissões mínimas necessárias para excluir um repositório que tenha a criptografia habilitada. A chave do KMS usada para criptografar o repositório pode ser especificada usando o parâmetro do recurso.

```
{
    "Version": "2012-10-17",
    "Id": "ecr-kms-permissions",
    "Statement": [
        {
            "Sid": "AllowAccessToRetireTheGrantsAssociatedWithTheKey",
            "Effect": "Allow",
            "Action": [
                "kms:RetireGrant"
            ],
            "Resource": "arn:aws:kms:us-
west-2:111122223333:key/b8d9ae76-080c-4043-92EXAMPLE"
        }
    ]
}
```

Permissões do IAM obrigatórias ao usar uma chave gerenciada pelo cliente

Ao criar um repositório com AWS KMS criptografia habilitada usando uma chave gerenciada pelo cliente, há permissões necessárias para a política de chaves do KMS e a política do IAM para o usuário ou função que está criando o repositório.

Ao criar sua própria chave KMS, você pode usar a política de chave padrão que o AWS KMS cria ou pode especificar o seu próprio valor. Para garantir que a chave gerenciada pelo cliente permaneça gerenciável pelo proprietário da conta, a política de chaves da chave KMS deve permitir todas as AWS KMS ações para o usuário raiz da conta. Permissões de escopo adicionais podem ser adicionadas à política de chave, mas pelo menos o usuário-raiz deve receber permissões para gerenciar a chave KMS. Para permitir que a chave KMS seja usada somente para solicitações originadas no Amazon ECR, você pode usar a chave de ViaService condição kms: com o valor. `ecr.<region>.amazonaws.com`

O exemplo de política de chaves a seguir dá à AWS conta (usuário raiz) que possui a chave KMS acesso total à chave KMS. Para obter mais informações sobre esse exemplo de política de chaves, consulte Permite acesso à AWS conta e ativa políticas do IAM no Guia do AWS Key Management Service desenvolvedor.

```
{
    "Version": "2012-10-17",
    "Id": "ecr-key-policy",
    "Statement": [
        {
            "Sid": "EnableIAMUserPermissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        }
    ]
}
```

O usuário do IAM, a função do IAM ou a AWS conta que está criando seus repositórios devem ter a `kms:DescribeKey` permissão `kms:CreateGrantkms:RetireGrant`, e, além das permissões necessárias do Amazon ECR.

> **Note**
>
> A permissão `kms:RetireGrant` deve ser adicionada à política do IAM do usuário ou função que cria o repositório. As permissões `kms:CreateGrant` e `kms:DescribeKey` podem ser adicionadas à política de chave para a chave KMS ou à política do IAM de usuário ou função que cria o repositório. Para obter mais informações sobre como AWS KMS as permissões

funcionam, consulte Permissões de AWS KMS API: referência de ações e recursos no Guia do AWS Key Management Service desenvolvedor.

O exemplo a seguir de política do IAM pode ser adicionado como uma política em linha a um usuário para garantir que ele tenha as permissões mínimas necessárias para criar um repositório com criptografia habilitada e excluir o repositório quando não precisar mais dele. A AWS KMS key usada para criptografar o repositório pode ser especificada usando o parâmetro do recurso.

```
{
    "Version": "2012-10-17",
    "Id": "ecr-kms-permissions",
    "Statement": [
        {
            "Sid":
 "AllowAccessToCreateAndRetireTheGrantsAssociatedWithTheKeyAsWellAsDescribeTheKey",
            "Effect": "Allow",
            "Action": [
                "kms:CreateGrant",
                "kms:RetireGrant",
                "kms:DescribeKey"
            ],
            "Resource": "arn:aws:kms:us-
west-2:111122223333:key/b8d9ae76-080c-4043-92EXAMPLE"
        }
    ]
}
```

Permitir que um usuário liste chaves KMS no console ao criar um repositório

Ao usar o console do Amazon ECR para criar um repositório, você pode conceder permissões para que um usuário liste as chaves KMS gerenciadas pelo cliente na região quando habilitar a criptografia para o repositório. O exemplo de política do IAM a seguir mostra as permissões necessárias para listar suas chaves e aliases do KMS ao usar o console.

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "kms:ListKeys",
```

```
      "kms:ListAliases",
      "kms:DescribeKey"
    ],
    "Resource": "*"
  }
}
```

Monitoramento da interação do Amazon ECR com o AWS KMS

Você pode usar AWS CloudTrail para rastrear as solicitações que o Amazon ECR envia AWS KMS em seu nome. As entradas de registro no CloudTrail registro contêm uma chave de contexto de criptografia para torná-las mais facilmente identificáveis.

Contexto de criptografia do Amazon ECR

Um contexto de criptografia é um conjunto de pares de chave/valor que contém dados arbitrários não secretos. Quando você inclui um contexto de criptografia em uma solicitação para criptografar dados, vincula AWS KMS criptograficamente o contexto de criptografia aos dados criptografados. Para descriptografar os dados, você deve passar o mesmo contexto de criptografia.

Em suas solicitações de GenerateDataKey and Decrypt, o AWS KMS Amazon ECR usa um contexto de criptografia com dois pares de nome e valor que identificam o repositório e o bucket do Amazon S3 que estão sendo usados. Isso é mostrado no exemplo a seguir. Os nomes não variam, mas os valores de contexto de criptografia combinados serão diferentes para cada valor.

```
"encryptionContext": {
    "aws:s3:arn": "arn:aws:s3:::us-west-2-starport-manifest-bucket/EXAMPLE1-90ab-cdef-fedc-ba987BUCKET1/
sha256:a7766145a775d39e53a713c75b6fd6d318740e70327aaa3ed5d09e0ef33fc3df",
    "aws:ecr:arn": "arn:aws:ecr:us-west-2:111122223333:repository/repository-name"
}
```

Você pode usar o contexto de criptografia para identificar essas operações criptográficas em registros e registros de auditoria, como AWS CloudTrailAmazon CloudWatch Logs, e como condição para autorização em políticas e concessões.

O contexto de criptografia do Amazon ECR consiste em dois pares de nome e valor.

- aws:s3:arn – O par de nome e valor identifica o bucket. A chave é `aws:s3:arn`. O valor de nome do recurso da Amazon (ARN) do bucket do Amazon S3

```
"aws:s3:arn": "ARN of an Amazon S3 bucket"
```

Por exemplo, se o ARN de um bucket fosse `arn:aws:s3:::us-west-2-starport-manifest-bucket/EXAMPLE1-90ab-cdef-fedc-ba987BUCKET1/sha256:a7766145a775d39e53a713c75b6fd6d318740e70327aaa3ed5d09e0ef33fc3df`, o contexto de criptografia incluiria o seguinte par.

```
"arn:aws:s3:::us-west-2-starport-manifest-bucket/EXAMPLE1-90ab-cdef-fedc-ba987BUCKET1/sha256:a7766145a775d39e53a713c75b6fd6d318740e70327aaa3ed5d09e0ef33fc3df"
```

- aws:ecr:arn – O segundo par de nome e valor identifica nome do recurso da Amazon (ARN) do repositório. A chave é `aws:ecr:arn`. O valor é o ARN do repositório.

  ```
  "aws:ecr:arn": "ARN of an Amazon ECR repository"
  ```

  Por exemplo, se o ARN do repositório fosse `arn:aws:ecr:us-west-2:111122223333:repository/repository-name`, o contexto de criptografia incluiria o seguinte par.

  ```
  "aws:ecr:arn": "arn:aws:ecr:us-west-2:111122223333:repository/repository-name"
  ```

Solução de problemas

Ao excluir um repositório do Amazon ECR com o console, se o repositório for excluído com sucesso, mas o Amazon ECR não conseguir retirar as concessões adicionadas à sua chave KMS para seu repositório, você receberá o seguinte erro.

```
The repository [{repository-name}] has been deleted successfully but the grants created
 by the kmsKey [{kms_key}] failed to be retired
```

Quando isso ocorrer, você mesmo poderá retirar as AWS KMS concessões do repositório.

Para retirar manualmente os AWS KMS subsídios de um repositório

1. Liste as concessões para a AWS KMS chave usada no repositório. O valor `key-id` é incluído no erro que você recebe do console. Você também pode usar o `list-keys` comando para listar as

chaves KMS gerenciadas pelo cliente Chaves gerenciadas pela AWS e as chaves do KMS em uma região específica da sua conta.

```
aws kms list-grants \
     --key-id b8d9ae76-080c-4043-9237-c815bfc21dfc
     --region us-west-2
```

A saída inclui um `EncryptionContextSubset` com o nome do recurso da Amazon (ARN) do seu repositório. Isso pode ser usado para determinar qual é a concessão adicionada à chave que você deseja retirar. O valor `GrantId` será usado quando for retirada a concessão na próxima etapa.

2. Retire cada concessão da AWS KMS chave adicionada ao repositório. Substitua o valor de pelo *`GrantId`*ID da concessão da saída da etapa anterior.

```
aws kms retire-grant \
     --key-id b8d9ae76-080c-4043-9237-c815bfc21dfc \
     --grant-id GrantId \
     --region us-west-2
```

Validação da conformidade do Amazon Elastic Container Registry

Para saber se um AWS service (Serviço da AWS) está dentro do escopo de programas de conformidade específicos, consulte Serviços da AWS Escopo por Programa de Conformidade Serviços da AWS e escolha o programa de conformidade em que você está interessado. Para obter informações gerais, consulte Programas de AWS conformidade Programas AWS de .

Você pode baixar relatórios de auditoria de terceiros usando AWS Artifact. Para obter mais informações, consulte Baixar relatórios em AWS Artifact .

Sua responsabilidade de conformidade ao usar Serviços da AWS é determinada pela confidencialidade de seus dados, pelos objetivos de conformidade de sua empresa e pelas leis e regulamentações aplicáveis. AWS fornece os seguintes recursos para ajudar na conformidade:

- Guias de início rápido sobre segurança e conformidade — Esses guias de implantação discutem considerações arquitetônicas e fornecem etapas para a implantação de ambientes básicos AWS focados em segurança e conformidade.

- Arquitetura para segurança e conformidade com a HIPAA na Amazon Web Services — Este whitepaper descreve como as empresas podem usar AWS para criar aplicativos qualificados para a HIPAA.

> **Note**
>
> Nem todos Serviços da AWS são elegíveis para a HIPAA. Para obter mais informações, consulte Referência dos Serviços Qualificados pela HIPAA.

- AWS Recursos de https://aws.amazon.com/compliance/resources/ de conformidade — Essa coleção de pastas de trabalho e guias pode ser aplicada ao seu setor e local.
- AWS Guias de conformidade do cliente — Entenda o modelo de responsabilidade compartilhada sob a ótica da conformidade. Os guias resumem as melhores práticas de proteção 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)).
- Avaliação de recursos com regras no Guia do AWS Config desenvolvedor — O AWS Config serviço avalia o quão bem suas configurações de recursos estão em conformidade com as práticas internas, as diretrizes e os regulamentos do setor.
- AWS Security Hub— Isso AWS service (Serviço da AWS) fornece uma visão abrangente do seu estado de segurança interno 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 com suporte e controles aceitos, consulte a Referência de controles do Security Hub.
- Amazon GuardDuty — Isso AWS service (Serviço da AWS) detecta possíveis ameaças às suas cargas de trabalho Contas da AWS, contêineres e dados monitorando seu ambiente em busca de atividades suspeitas e maliciosas. GuardDuty pode ajudá-lo a atender a vários requisitos de conformidade, como o PCI DSS, atendendo aos requisitos de detecção de intrusões exigidos por determinadas estruturas de conformidade.
- AWS Audit Manager— Isso AWS service (Serviço da AWS) ajuda você a auditar continuamente seu AWS uso para simplificar a forma como você gerencia o risco e a conformidade com as regulamentações e os padrões do setor.

Segurança de infraestrutura no Amazon Elastic Container Registry

Como um serviço gerenciado, o Amazon Elastic Container Registry é protegido pela segurança de rede AWS global. Para obter informações sobre serviços AWS de segurança e como AWS proteger a infraestrutura, consulte AWS Cloud Security. Para projetar seu AWS ambiente usando as melhores práticas de segurança de infraestrutura, consulte Proteção de infraestrutura no Security Pillar AWS Well-Architected Framework.

Você usa chamadas de API AWS publicadas para acessar o Amazon ECR pela rede. Os clientes devem ser compatíveis com:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Conjuntos de criptografia com Perfect Forward Secrecy (PFS) como DHE (Ephemeral Diffie-Hellman) ou ECDHE (Ephemeral Elliptic Curve Diffie-Hellman). A maioria dos sistemas modernos, como Java 7 e versões posteriores, suporta 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 é possível usar o AWS Security Token Service (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

É possível chamar essas operações de API de qualquer local da rede, mas o Amazon ECR não é compatível com políticas de acesso baseadas em recursos, que podem incluir restrições com base no endereço IP de origem. Também é possível usar políticas do Amazon ECR para controlar o acesso a partir de endpoints da Amazon Virtual Private Cloud (Amazon VPC) ou de VPCs específicos. Efetivamente, isso isola o acesso à rede a um determinado recurso do Amazon ECR somente da VPC específica dentro da rede. AWS Para ter mais informações, consulte Endpoints VPC da interface Amazon ECR ()AWS PrivateLink.

Endpoints VPC da interface Amazon ECR ()AWS PrivateLink

É possível melhorar a postura de segurança da sua VPC configurando o Amazon ECR para usar um endpoint de interface da VPC. Os VPC endpoints são alimentados por AWS PrivateLink, uma tecnologia que permite que você acesse de forma privada as APIs do Amazon ECR por meio de endereços IP privados. AWS PrivateLink restringe todo o tráfego de rede entre sua VPC e o Amazon ECR para a rede Amazon. Você não precisa de um gateway da Internet, de um dispositivo NAT ou de um gateway privado virtual.

Para obter mais informações sobre AWS PrivateLink VPC endpoints, consulte VPC Endpoints no Guia do usuário da Amazon VPC.

Considerações endpoints da VPC do Amazon ECR

Antes de configurar endpoints da VPC para o Amazon ECR, fique atento às seguintes considerações:

- Para permitir que suas tarefas do Amazon ECS hospedadas nas instâncias do Amazon EC2 extraiam imagens privadas do Amazon ECR, crie também endpoints de interface da VPC para o Amazon ECS. Para obter mais informações, consulte Interface VPC Endpoints (AWS PrivateLink) no Amazon Elastic Container Service Developer Guide.

  > **Important**
  >
  > As tarefas do Amazon ECS hospedadas no Fargate não exigem os endpoints de interface da VPC do Amazon ECS.

- As tarefas do Amazon ECS hospedadas no Fargate que usam a plataforma Linix versão `1.3.0` ou anterior exigem apenas o endpoint da VPC do Amazon ECR com.amazonaws.***region***.ecr.dkr e o endpoint do gateway do Amazon S3 para aproveitar esse recurso.
- As tarefas do Amazon ECS hospedadas no Fargate que usam a plataforma Linux versão `1.4.0` ou posterior exigem tanto os endpoints da VPC do Amazon ECR com.amazonaws.***region***.ecr.dkr e com.amazonaws.***region***.ecr.api quanto o endpoint do gateway do Simple Storage Service (Amazon S3) para poderem aproveitar esse recurso.
- As tarefas do Amazon ECS hospedadas no Fargate que usam a plataforma Windows versão `1.0.0` ou posterior exigem tanto os endpoints da VPC do Amazon ECR com.amazonaws.***region***.ecr.dkr e com.amazonaws.***region***.ecr.api quanto o endpoint do gateway do Simple Storage Service (Amazon S3) para poderem aproveitar esse recurso.
- As tarefas do Amazon ECS hospedadas no Fargate que extraem imagens do Amazon ECR podem restringir o acesso à VPC específica que as tarefas usam e ao endpoint da VPC que o serviço usa, adicionando chaves de condição à função do IAM para a tarefa. Para obter mais informações, consulte Permissões opcionais do IAM para tarefas do Fargate que extraem imagens do Amazon ECR por endpoints de interface no Guia do desenvolvedor do Amazon Elastic Container Service.
- As tarefas do Amazon ECS hospedadas no Fargate que extraem imagens de contêineres do Amazon ECR que também usam `awslogs` o driver de log para enviar informações CloudWatch de

log para a Logs exigem o endpoint VPC do CloudWatch Logs. Para ter mais informações, consulte Crie o endpoint do CloudWatch Logs.

- O grupo de segurança anexado ao endpoint da VPC deve permitir conexões de entrada na porta 443 na sub-rede privada da VPC.
- Atualmente, os endpoints da VPC não oferecem suporte a solicitações entre Regiões. Certifique-se de criar endpoints da VPC na mesma região em que você planeja emitir chamadas de API para o Amazon ECR.
- No momento, os endpoints da VPC não oferecem suporte aos repositórios públicos do Amazon ECR. Considere usar uma regra de cache de pull through para hospedar a imagem pública em um repositório privado na mesma região do endpoint da VPC. Para ter mais informações, consulte Sincronize um registro upstream com um registro privado do Amazon ECR.
- Os VPC endpoints oferecem suporte somente ao DNS AWS fornecido por meio do Amazon Route 53. Se quiser usar seu próprio DNS, você pode usar o encaminhamento de DNS condicional. Para obter mais informações, consulte Conjuntos de opções de DHCP no Manual do usuário da Amazon VPC.
- Se os contêineres tiverem conexões existentes com o Amazon S3, as conexões poderão ser interrompidas brevemente quando você adicionar o endpoint do gateway do Amazon S3. Se você quiser evitar essa interrupção, crie uma VPC que usa o endpoint de gateway do Amazon S3 e migre o cluster do Amazon ECS e seus contêineres para a nova VPC.
- Quando uma imagem é extraída usando uma regra de cache de pull-through pela primeira vez, se você configurou o Amazon ECR para usar um endpoint de VPC de interface usando AWS PrivateLink , então é necessário criar uma sub-rede pública na mesma VPC, com um gateway NAT e, em seguida, rotear todo o tráfego de saída para a Internet de sua sub-rede privada para o gateway NAT para que a extração funcione. As extrações de imagem subsequentes não exigem isso. Para obter mais informações, consulte Cenário: Acessar a Internet de uma sub-rede privada no Guia do usuário da Amazon Virtual Private Cloud.

Considerações para imagens do Windows

As imagens baseadas no sistema operacional Windows incluem artefatos que são restringidos, pela licença, de serem distribuídos. Por padrão, quando você envia imagens do Windows para um repositório do Amazon ECR, as camadas que incluem esses artefatos não são enviadas, pois elas são consideradas camadas externas. Quando os artefatos são fornecidos pela Microsoft, as camadas externas são recuperadas da infraestrutura do Microsoft Azure. Por esse motivo, para

permitir que seus contêineres extraiam essas camadas externas do Azure, etapas adicionais são necessárias além da criação dos endpoints da VPC.

É possível substituir esse comportamento ao enviar imagens do Windows ao Amazon ECR usando o sinalizador `--allow-nondistributable-artifacts` no daemon do Docker. Quando habilitado, esse sinalizador envia as camadas licenciadas para o Amazon ECR, o que permite que essas imagens sejam extraídas do Amazon ECR por meio do endpoint da VPC sem necessidade de acesso adicional ao Azure.

> **Important**
>
> Usar o sinalizador `--allow-nondistributable-artifacts` não exclui sua obrigação de cumprir os termos da licença de imagem base de contêiner do Windows. Você não pode postar conteúdo do Windows para redistribuição pública ou de terceiros. O uso dentro do seu próprio ambiente é permitido.

Para habilitar o uso desse sinalizador para a instalação do Docker, você deve modificar o arquivo de configuração do daemon do Docker que, dependendo da instalação do Docker, normalmente pode ser configurado no menu de configurações ou preferências na seção Engine do Docker ou editando a seção do arquivo `C:\ProgramData\docker\config\daemon.json` diretamente.

Veja a seguir um exemplo da configuração necessária: Substitua o valor pelo URI do repositório para o qual você está enviando imagens.

```
{
    "allow-nondistributable-artifacts": [
        "111122223333.dkr.ecr.us-west-2.amazonaws.com"
    ]
}
```

Depois de modificar o arquivo de configuração do daemon do Docker, você deve reiniciar o daemon do Docker antes de tentar enviar sua imagem. Confirme se o push funcionou verificando se a camada base foi enviada ao seu repositório.

> **Note**
>
> As camadas base para imagens do Windows são grandes. O tamanho da camada resulta em mais tempo de envio e custos adicionais de armazenamento no Amazon ECR para

essas imagens. Por esses motivos, recomendamos usar essa opção somente quando for estritamente necessário reduzir os tempos de construção e os custos contínuos de armazenamento. Por exemplo, a imagem `mcr.microsoft.com/windows/servercore` tem aproximadamente 1,7 GiB de tamanho quando compactada no Amazon ECR.

Criação dos endpoints da VPC para o Amazon ECR

Para criar os endpoints da VPC para o serviço do Amazon ECR, use o procedimento de Criação de um endpoint de interface no Manual do usuário da Amazon VPC.

As tarefas do Amazon ECS hospedadas em instâncias do Amazon EC2 exigem endpoints do Amazon ECR e o endpoint do gateway do Amazon S3.

As Tarefas do Amazon ECS hospedadas no Fargate usando a versão `1.4.0` da plataforma ou posterior exigem endpoints da VPC do Amazon ECR e endpoints de gateway do Amazon S3.

As tarefas do Amazon ECS hospedadas no Fargate usando a versão `1.3.0` ou anterior da plataforma exigem apenas os endpoints da VPC do Amazon ECR com.amazonaws.***region***.ecr.dkr e o endpoint da VPC do Amazon ECR e os endpoints de gateway do Amazon S3 para aproveitar esse recurso.

> **Note**
>
> A ordem em que os endpoints são criados não importa.

com.amazonaws.***region***.ecr.dkr

Esse endpoint é usado para as APIs de registro do Docker. Os comandos de cliente do Docker, como `push` e `pull`, usam esse endpoint.

Ao criar esse endpoint, você deve habilitar um nome de host DNS privado. Para fazer isso, verifique se a opção Enable Private DNS Name (Habilitar nome DNS privado) está selecionada no console da Amazon VPC ao criar o endpoint da VPC.

com.amazonaws.*region*.ecr.api

> **Note**
>
> A *região* especificada representa o identificador de uma AWS região suportada pelo Amazon ECR, como `us-east-2` a região Leste dos EUA (Ohio).

Esse endpoint é usado para chamadas à API do Amazon ECR. As opções da API, como `DescribeImages` e `CreateRepository` são enviadas para esse endpoint.

Quando esse endpoint é criado, você tem a opção de habilitar um nome de host DNS privado. Habilite essa configuração selecionando Habilitar nome DNS privado no console da VPC ao criar o VPC endpoint. Se você habilitar um nome de host DNS privado para o VPC endpoint, atualize seu SDK ou AWS CLI para a versão mais recente para que não seja necessário especificar uma URL de endpoint ao usar o SDK ou não. AWS CLI

Se você habilitar um nome de host DNS privado e estiver usando um SDK ou uma AWS CLI versão lançada antes de 24 de janeiro de 2019, deverá usar o `--endpoint-url` parâmetro para especificar os endpoints da interface. O exemplo a seguir mostra o formato do URL do endpoint.

```
aws ecr create-repository --repository-name name --endpoint-url https://
api.ecr.region.amazonaws.com
```

Se você não habilitar um nome de host DNS privado para o VPC endpoint, deverá usar o parâmetro `--endpoint-url` especificando o ID do VPC endpoint para o endpoint de interface. O exemplo a seguir mostra o formato do URL do endpoint.

```
aws ecr create-repository --repository-name name --endpoint-url
 https://VPC_endpoint_ID.api.ecr.region.vpce.amazonaws.com
```

Criar o endpoint do gateway do Amazon S3

Para que as tarefas do Amazon ECS extraiam imagens privadas do Amazon ECR, crie um endpoint de gateway para o Amazon S3. O endpoint do gateway é necessário porque o usa o Amazon ECR usa o Amazon S3 para armazenar as camadas de imagem. Quando os contêineres baixam imagens do Amazon ECR, eles devem acessar o Amazon ECR para obter o manifesto da imagem e

o Amazon S3 para baixar as camadas reais da imagem. Este é o nome do recurso da Amazon (ARN) do bucket do Amazon S3 que contém as camadas de cada imagem do Docker.

```
arn:aws:s3:::prod-region-starport-layer-bucket/*
```

Usar o procedimento Criação de um endpoint de gateway no Manual do usuário da Amazon VPC para criar o seguinte endpoint de gateway do Amazon S3 para o Amazon ECR. Ao criar o endpoint, selecione as tabelas de rotas para sua VPC.

com.amazonaws.***region***.s3

> O endpoint de gateway do Amazon S3 usa um documento de política do IAM para limitar o acesso ao serviço. A política de Acesso total pode ser usada, pois qualquer restrição colocada nas funções do IAM de sua tarefa ou outras políticas de usuário do IAM ainda serão aplicadas além dessa política. Se você quiser limitar o acesso ao bucket do Amazon S3 para as permissões mínimas exigidas para usar o Amazon ECR, consulte Permissões mínimas do bucket do Amazon S3 para o Amazon ECR.

Permissões mínimas do bucket do Amazon S3 para o Amazon ECR

O endpoint de gateway do Amazon S3 usa um documento de política do IAM para limitar o acesso ao serviço. Para conceder apenas as permissões mínimas do bucket do Amazon S3 para o Amazon ECR, restrinja o acesso ao bucket do Amazon S3 usado pelo Amazon ECR ao criar o documento de política do IAM para o endpoint.

A tabela a seguir descreve as permissões de política do bucket do Amazon S3 exigidas pelo Amazon ECR.

Permissão	Descrição
`arn:aws:s3:::prod-region-starport-layer-bucket/*`	Fornece acesso ao bucket do Amazon S3 que contém as camadas de cada imagem do Docker. Representa o identificador da região para uma região da AWS suportada pelo Amazon ECR, como `us-east-2` para a região Leste dos EUA (Ohio).

Exemplo

O exemplo a seguir ilustra como conceder acesso aos buckets do Amazon S3 exigidos para as operações do Amazon ECR.

```
{
  "Statement": [
    {
      "Sid": "Access-to-specific-bucket-only",
      "Principal": "*",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": ["arn:aws:s3:::prod-region-starport-layer-bucket/*"]
    }
  ]
}
```

Crie o endpoint do CloudWatch Logs

As tarefas do Amazon ECS que usam o tipo de execução Fargate que usam uma VPC sem um gateway de internet e que também usam **`awslogs`** o driver de log para enviar informações de log para o Logs exigem que CloudWatch você crie o com.amazonaws. interface ***`region`*** .logs VPC endpoint CloudWatch para Logs. Para obter mais informações, consulte Como usar CloudWatch registros com endpoints VPC de interface no Guia do usuário do Amazon CloudWatch Logs.

Criar uma política de endpoint para os endpoints da VPC do Amazon ECR

Uma política de endpoint da VPC é uma política de recursos do IAM que você anexa a um endpoint quando cria ou modifica o endpoint. Se você não anexar uma política ao criar um endpoint, AWS anexará uma política padrão que permita acesso total ao serviço. Uma política de endpoint não substitui políticas de usuário do ou políticas de serviço específicas. É uma política separada para controlar o acesso do endpoint ao serviço especificado. Políticas de endpoint devem ser gravadas em formato JSON. Para obter mais informações, consulte Controlar o acesso a serviços com VPC endpoints no Manual do usuário da Amazon VPC.

Recomendamos criar uma única política de recursos do IAM e anexá-la a ambos os endpoints da VPC do Amazon ECR.

A seguir temos um exemplo de uma política de endpoint para o Amazon ECR Essa política permite que uma função do IAM específica extraia imagens do Amazon ECR.

```
{
 "Statement": [{
  "Sid": "AllowPull",
  "Principal": {
   "AWS": "arn:aws:iam::1234567890:role/role_name"
  },
  "Action": [
   "ecr:BatchGetImage",
   "ecr:GetDownloadUrlForLayer",
                  "ecr:GetAuthorizationToken"
  ],
  "Effect": "Allow",
  "Resource": "*"
 }]
}
```

O exemplo de política de endpoint a seguir impede que um repositório especificado seja excluído.

```
{
 "Statement": [{
   "Sid": "AllowAll",
   "Principal": "*",
   "Action": "*",
   "Effect": "Allow",
   "Resource": "*"
  },
  {
   "Sid": "PreventDelete",
   "Principal": "*",
   "Action": "ecr:DeleteRepository",
   "Effect": "Deny",
   "Resource": "arn:aws:ecr:region:1234567890:repository/repository_name"
  }
 ]
}
```

O exemplo de política de endpoint a seguir combina os dois exemplos anteriores em uma única política.

```
{
 "Statement": [{
   "Sid": "AllowAll",
   "Effect": "Allow",
   "Principal": "*",
   "Action": "*",
   "Resource": "*"
  },
  {
   "Sid": "PreventDelete",
   "Effect": "Deny",
   "Principal": "*",
   "Action": "ecr:DeleteRepository",
   "Resource": "arn:aws:ecr:region:1234567890:repository/repository_name"
  },
  {
   "Sid": "AllowPull",
   "Effect": "Allow",
   "Principal": {
    "AWS": "arn:aws:iam::1234567890:role/role_name"
   },
   "Action": [
    "ecr:BatchGetImage",
    "ecr:GetDownloadUrlForLayer",
                         "ecr:GetAuthorizationToken"
   ],
   "Resource": "*"
  }
 ]
}
```

Para modificar a política endpoint da VPC para o Amazon ECR

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Endpoints.
3. Se você ainda não criou os endpoints da VPC para o Amazon ECR, consulte Criação dos endpoints da VPC para o Amazon ECR.
4. Selecione endpoint da VPC do Amazon ECR ao qual deseja adicionar uma política e escolha a guia Policy (Política) na parte inferior da tela.
5. Selecione Edit policy (Editar política) e faça as alterações na política.

6. Escolha Save (Salvar) para salvar a política.

Sub-redes compartilhadas

Você não pode criar, descrever, modificar ou excluir endpoints da VPC em sub-redes que são compartilhadas com você. No entanto, você pode usar os endpoints da VPC em sub-redes que são compartilhadas com você.

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

O problema “confused deputy” é um problema de segurança no qual uma entidade que não tem permissão para executar uma ação pode coagir uma entidade mais privilegiada a executá-la. Em AWS, a falsificação de identidade entre serviços pode resultar no problema confuso do deputado. 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 a usar suas permissões para atuar nos recursos de outro cliente indo contra permissão de acesso. 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 Amazon ECR concede a outro serviço no recurso para o recurso. Use `aws:SourceArn` se quiser apenas um recurso associado a acessibilidade de serviço. 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 especificar vários recursos, use a chave de condição de contexto global `aws:SourceArn` com caracteres curinga (*) para as partes desconhecidas do ARN. Por exemplo, .`arn:aws:`*`servicename`*`:`*`region`*`:`*`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 ResourceDescription.

O exemplo a seguir mostra como você pode usar as chaves de contexto de condição `aws:SourceAccount` global `aws:SourceArn` e as chaves de contexto em uma política de repositório do Amazon ECR para permitir o AWS CodeBuild acesso às ações de API do Amazon ECR necessárias para a integração com esse serviço e, ao mesmo tempo, evitar o problema confuso do substituto.

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"CodeBuildAccess",
         "Effect":"Allow",
         "Principal":{
            "Service":"codebuild.amazonaws.com"
         },
         "Action":[
            "ecr:BatchGetImage",
            "ecr:GetDownloadUrlForLayer"
         ],
         "Condition":{
            "ArnLike":{
               "aws:SourceArn":"arn:aws:codebuild:region:123456789012:project/project-
name"
            },
            "StringEquals":{
               "aws:SourceAccount":"123456789012"
            }
         }
      }
   ]
}
```

Monitoramento do Amazon ECR

Você pode monitorar o uso da API do Amazon ECR com a Amazon CloudWatch, que coleta e processa dados brutos do Amazon ECR em métricas legíveis e quase em tempo real. Essas estatísticas são registradas por um período de duas semanas para que você possa acessar informações históricas e ter uma perspectiva sobre o uso da API. Os dados métricos do Amazon ECR são enviados automaticamente CloudWatch em períodos de um minuto. Para obter mais informações sobre CloudWatch, consulte o Guia CloudWatch do usuário da Amazon.

O Amazon ECR fornece métricas com base no uso de API para ações de autorização, envio de imagem e extração de imagem.

O monitoramento é uma parte importante da manutenção da confiabilidade, disponibilidade e desempenho do Amazon ECR e de suas AWS soluções. Recomendamos que você colete dados de monitoramento dos recursos que compõem sua AWS solução para poder depurar com mais facilidade uma falha de vários pontos, caso ocorra. No entanto, antes de começar a monitorar o Amazon ECR, é necessário criar um plano de monitoramento que inclua respostas às seguintes perguntas:

- Quais são seus objetivos de monitoramento?
- Quais recursos você vai monitorar?
- Com que frequência você vai monitorar esses recursos?
- Quais ferramentas de monitoramento você usará?
- Quem realizará o monitoramento das tarefas?
- Quem deve ser notificado quando algo der errado?

A próxima etapa é estabelecer uma linha de base de performance normal do Amazon ECR no seu ambiente, medindo o performance em vários momentos e em diferentes condições de carga. À medida que você monitorar o Amazon ECR, armazene dados de monitoramento históricos para compará-los com os novos dados de performance, identificar padrões de performance normais e anomalias de performance, além de elaborar métodos para resolver problemas.

Tópicos

- Relatórios de uso do Amazon ECR
- Métricas do repositório do Amazon ECR
- Eventos do Amazon ECR e EventBridge
- Registrando ações do Amazon ECR com AWS CloudTrail

Visualizar as Service Quotas e definir alarmes

Você pode usar o CloudWatch console para visualizar suas cotas de serviço e ver como seu uso atual se compara às cotas de serviço. Também é possível definir alarmes para que você seja notificado ao se aproximar de uma cota.

Como visualizar uma cota de serviço e opcionalmente definir um alarme

1. Abra o CloudWatch console em https://console.aws.amazon.com/cloudwatch/.
2. No painel de navegação, selecione Métricas.
3. Na guia Todas as métricas, escolha Uso e depois Por recurso da AWS .

   A lista das métricas de uso da cota de serviço é exibida.
4. Marque a caixa de seleção ao lado de uma das métricas.

   O gráfico mostra seu uso atual desse AWS recurso.
5. Para adicionar a cota de serviço ao gráfico, faça o seguinte:

   a. Escolha a guia Graphed metrics (Métricas em gráfico).

   b. Selecione Math expression (Expressão matemática), Start with an empty expression (Começar com uma expressão vazia). Depois, na nova linha, em Details (Detalhes), insira **`SERVICE_QUOTA(m1)`**.

      Uma nova linha é adicionada ao gráfico, exibindo a cota de serviço do recurso representado na métrica.
6. Para ver o uso atual como uma porcentagem da cota, adicione uma nova expressão ou altere a expressão SERVICE_QUOTA atual. Para a nova expressão, use **`m1/60/SERVICE_QUOTA(m1)*100`**.
7. (Opcional) Para definir um alarme que notifique se você caso se aproxime da cota de serviço, faça o seguinte:

a. Na linha **`m1/60/SERVICE_QUOTA(m1)*100`**, em Actions (Ações), selecione o ícone de alarme. Ele se parece com um sino.

   A página de criação de alarmes é exibida.

b. Em Conditions (Condições), verifique se o Threshold type (Tipo de limite) é Static (Estático) e se Whenever Expression1 is (Sempre que a Expression1 for) esteja definido como Greater (Maior). Em than (que), insira **`80`**. Isso cria um alarme que entrará no estado ALARM (ALARME) quando seu uso exceder 80% da cota.

c. Escolha Próximo.

d. Na próxima página, selecione um tópico do Amazon SNS ou crie um. Esse tópico será notificado quando o alarme entrar no estado ALARM (ALARME). Em seguida, escolha Próximo.

e. Na próxima página, insira um nome e uma descrição para o alarme e selecione Next (Próximo).

f. Selecione Criar alarme.

Métricas de uso do Amazon ECR

Você pode usar métricas de CloudWatch uso para dar visibilidade ao uso dos recursos da sua conta. Use essas métricas para visualizar seu uso atual do serviço em CloudWatch gráficos e painéis.

As métricas de uso do Amazon ECR correspondem às cotas AWS de serviço. Também é possível configurar alarmes que alertem você quando o uso se aproximar de uma cota de serviço. Para obter mais informações sobre cotas de serviço do Amazon ECR, consulte Cotas de serviço do Amazon ECR.

O Amazon ECR publica as seguintes métricas no namespace `AWS/Usage`.

Métrica	Descrição
`CallCount`	O número de chamadas de ação de API da sua conta. Os recursos são definidos pelas dimensões associadas à métrica. A estatística mais útil para essa métrica é `SUM`, que representa a soma dos valores de todos os colaboradores durante o período definido.

As dimensões a seguir são usadas para refinar as métricas de uso publicadas pelo Amazon ECR.

Dimensão	Descrição
`Service`	O nome do AWS serviço que contém o recurso. Para as métricas de uso do Amazon ECR, o valor dessa dimensão é `ECR`.
`Type`	O tipo de entidade que está sendo relatado. Atualmente, o único valor válido para métricas de uso do Amazon ECR é `API`.
`Resource`	O tipo de recurso que está em execução. No momento, o Amazon ECR retorna informações sobre o uso da API para as ações de API a seguir. • GetAuthorizationToken • BatchCheckLayerAvailability • InitiateLayerUpload • UploadLayerPart • CompleteLayerUpload • PutImage • BatchGetImage • GetDownloadUrlForLayer
`Class`	A classe do recurso sob acompanhamento. No momento, o Amazon ECR não usa a dimensão de classe.

Relatórios de uso do Amazon ECR

AWS fornece uma ferramenta de geração de relatórios gratuita chamada Cost Explorer, que permite analisar o custo e o uso dos recursos do Amazon ECR.

Use o Cost Explorer para visualizar gráficos de uso e de custos. É possível visualizar dados dos últimos 13 meses e prever o valor que você provavelmente gastará nos próximos três meses. É possível usar o Cost Explorer para ver padrões de gastos de recursos da AWS ao longo do tempo, identificar áreas que precisam de uma investigação mais profunda e ver tendências que é possível

usar para entender seus custos. Também é possível especificar os períodos dos dados e visualizar os dados de tempo por dia ou por mês.

Os dados de medição nos Relatórios de uso e de custo mostram o uso em todos os repositórios do Amazon ECR. Para ter mais informações, consulte Marcar recursos para faturamento.

Para obter mais informações sobre a criação de um relatório de AWS custo e uso, consulte Relatório de AWS custo e uso no Guia AWS Billing do usuário.

Métricas do repositório do Amazon ECR

O Amazon ECR envia métricas de contagem de pull do repositório para a Amazon. CloudWatch Os dados métricos do Amazon ECR são enviados automaticamente CloudWatch em períodos de 1 minuto. Para obter mais informações sobre CloudWatch, consulte o Guia CloudWatch do usuário da Amazon.

Tópicos

- Habilitando CloudWatch métricas
- Métricas e dimensões disponíveis
- Visualizando métricas do Amazon ECR usando o console CloudWatch

Habilitando CloudWatch métricas

O Amazon ECR envia métricas de repositório automaticamente para todos os repositórios. Não é preciso realizar nenhuma etapa manual.

Métricas e dimensões disponíveis

As seções a seguir listam as métricas e dimensões que o Amazon ECR envia para a Amazon CloudWatch.

Métricas do Amazon ECR

O Amazon ECR fornece métricas para você monitorar seus repositórios. Você pode medir o número de solicitações pull.

O namespace `AWS/ECR` inclui as métricas a seguir.

`RepositoryPullCount`

O número total de solicitações pull das imagens no repositório.

Dimensões válidas: `RepositoryName`.

Estatísticas válidas: média, mínima, máxima, soma, contagem de exemplo. A estatística mais útil é Sum.

Unit: Integer.

Dimensões para métricas do Amazon ECR

As métricas do Amazon ECR usam o namespace `AWS/ECR` e fornecem métricas para as dimensões a seguir.

`RepositoryName`

Essa dimensão filtra os dados solicitados para todas as imagens do contêiner em um repositório especificado.

Visualizando métricas do Amazon ECR usando o console CloudWatch

Você pode visualizar as métricas do repositório Amazon ECR no CloudWatch console. O CloudWatch console fornece uma exibição refinada e personalizável de seus recursos. Para obter mais informações, consulte o Guia CloudWatch do usuário da Amazon.

Para visualizar métricas no CloudWatch console

1. Abra o CloudWatch console em https://console.aws.amazon.com/cloudwatch/.
2. Na navegação à esquerda, escolha Metrics (Métricas), All metrics (Todas as métricas).
3. Na guia Procurar, em Namespaces da AWS , escolha ECR.
4. Escolha as métricas a serem exibidas. As métricas do repositório têm como escopo ECR > Métricas do repositório.

Eventos do Amazon ECR e EventBridge

A Amazon EventBridge permite que você automatize seus AWS serviços e responda automaticamente a eventos do sistema, como problemas de disponibilidade de aplicativos ou

alterações de recursos. Os eventos dos AWS serviços são entregues quase EventBridge em tempo real. É possível escrever regras simples para indicar quais eventos são do seu interesse e incluir ações automatizadas que deverão ser realizadas quando um evento corresponder à regra. Ações que podem ser automaticamente acionadas incluem:

- Adicionar eventos a grupos de CloudWatch registros no Logs
- Invocando uma função AWS Lambda
- Invocar o comando de execução do Amazon EC2
- Transmitir o evento Amazon Kinesis Data Streams
- Ativando uma máquina de AWS Step Functions estado
- Notificar um tópico do Amazon SNS ou uma fila do Amazon SQS

Para obter mais informações, consulte Getting Started with Amazon EventBridge no Guia EventBridge do usuário da Amazon.

Amostra de eventos do Amazon ECR

Veja a seguir exemplos de eventos do Amazon ECR. Os eventos são emitidos com base no melhor esforço.

Evento para um envio de imagem concluído

O evento a seguir é enviado quando cada envio de imagem é concluído. Para ter mais informações, consulte Enviando uma imagem do Docker para um repositório privado do Amazon ECR.

```
{
    "version": "0",
    "id": "13cde686-328b-6117-af20-0e5566167482",
    "detail-type": "ECR Image Action",
    "source": "aws.ecr",
    "account": "123456789012",
    "time": "2019-11-16T01:54:34Z",
    "region": "us-west-2",
    "resources": [],
    "detail": {
        "result": "SUCCESS",
        "repository-name": "my-repository-name",
        "image-digest":
 "sha256:7f5b2640fe6fb4f46592dfd3410c4a79dac4f89e4782432e0378abcd1234",
```

```
        "action-type": "PUSH",
        "image-tag": "latest"
    }
}
```

Evento para uma ação de cache de pull-through

Quando uma tentativa de ação de cache de pull-through é feita, o seguinte evento é enviado. Para ter mais informações, consulte Sincronize um registro upstream com um registro privado do Amazon ECR.

```
{
    "version": "0",
    "id": "85fc3613-e913-7fc4-a80c-a3753e4aa9ae",
    "detail-type": "ECR Pull Through Cache Action",
    "source": "aws.ecr",
    "account": "123456789012",
    "time": "2023-02-29T02:36:48Z",
    "region": "us-west-2",
    "resources": [
        "arn:aws:ecr:us-west-2:123456789012:repository/docker-hub/alpine"
    ],
    "detail": {
        "rule-version": "1",
        "sync-status": "SUCCESS",
        "ecr-repository-prefix": "docker-hub",
        "repository-name": "docker-hub/alpine",
        "upstream-registry-url": "public.ecr.aws",
        "image-tag": "3.17.2",
        "image-digest":
 "sha256:4aa08ef415aecc80814cb42fa41b658480779d80c77ab15EXAMPLE",
    }
}
```

Evento para uma verificação de imagem concluída (verificação básica)

Quando a verificação básica está habilitada para seu registro, o evento a seguir é enviado quando cada verificação de imagem é concluída. O parâmetro `finding-severity-counts` só retornará um valor de um nível de gravidade se existir algum. Por exemplo, se a imagem não contiver descobertas no nível `CRITICAL`, não será retornada uma contagem crítica. Para ter mais informações, consulte Escaneie imagens em busca de vulnerabilidades do sistema operacional no Amazon ECR.

> **Note**
>
> Para obter detalhes sobre eventos que o Amazon Inspector emite quando a verificação avançada está habilitada, consulteEventBridge eventos enviados para digitalização aprimorada no Amazon ECR.

```
{
    "version": "0",
    "id": "85fc3613-e913-7fc4-a80c-a3753e4aa9ae",
    "detail-type": "ECR Image Scan",
    "source": "aws.ecr",
    "account": "123456789012",
    "time": "2019-10-29T02:36:48Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ecr:us-east-1:123456789012:repository/my-repository-name"
    ],
    "detail": {
        "scan-status": "COMPLETE",
        "repository-name": "my-repository-name",
        "finding-severity-counts": {
        "CRITICAL": 10,
        "MEDIUM": 9
      },
        "image-digest":
 "sha256:7f5b2640fe6fb4f46592dfd3410c4a79dac4f89e4782432e0378abcd1234",
        "image-tags": []
    }
}
```

Evento para uma notificação de alteração em um recurso com verificação avançada habilitada (verificação avançada)

Quando a verificação avançada é habilitada para seu registro, o evento a seguir é enviado pelo Amazon ECR quando há uma alteração em um recurso que tem a verificação avançada habilitada. Isso inclui novos repositórios sendo criados, a frequência de verificação de um repositório sendo alterada ou quando as imagens são criadas ou excluídas em repositórios com a verificação avançada ativada. Para ter mais informações, consulte Digitalize imagens em busca de vulnerabilidades de software no Amazon ECR.

```
{
 "version": "0",
 "id": "0c18352a-a4d4-6853-ef53-0ab8638973bf",
 "detail-type": "ECR Scan Resource Change",
 "source": "aws.ecr",
 "account": "123456789012",
 "time": "2021-10-14T20:53:46Z",
 "region": "us-east-1",
 "resources": [],
 "detail": {
  "action-type": "SCAN_FREQUENCY_CHANGE",
  "repositories": [{
    "repository-name": "repository-1",
    "repository-arn": "arn:aws:ecr:us-east-1:123456789012:repository/repository-1",
    "scan-frequency": "SCAN_ON_PUSH",
    "previous-scan-frequency": "MANUAL"
   },
   {
    "repository-name": "repository-2",
    "repository-arn": "arn:aws:ecr:us-east-1:123456789012:repository/repository-2",
    "scan-frequency": "CONTINUOUS_SCAN",
    "previous-scan-frequency": "SCAN_ON_PUSH"
   },
   {
    "repository-name": "repository-3",
    "repository-arn": "arn:aws:ecr:us-east-1:123456789012:repository/repository-3",
    "scan-frequency": "CONTINUOUS_SCAN",
    "previous-scan-frequency": "SCAN_ON_PUSH"
   }
  ],
  "resource-type": "REPOSITORY",
  "scan-type": "ENHANCED"
 }
}
```

Evento para uma exclusão de imagem

O evento a seguir é enviado quando uma imagem é excluída. Para ter mais informações, consulte Excluindo uma imagem no Amazon ECR.

```
{
    "version": "0",
    "id": "dd3b46cb-2c74-f49e-393b-28286b67279d",
```

```
    "detail-type": "ECR Image Action",
    "source": "aws.ecr",
    "account": "123456789012",
    "time": "2019-11-16T02:01:05Z",
    "region": "us-west-2",
    "resources": [],
    "detail": {
        "result": "SUCCESS",
        "repository-name": "my-repository-name",
        "image-digest":
 "sha256:7f5b2640fe6fb4f46592dfd3410c4a79dac4f89e4782432e0378abcd1234",
        "action-type": "DELETE",
        "image-tag": "latest"
    }
}
```

Registrando ações do Amazon ECR com AWS CloudTrail

O Amazon ECR é integrado com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, uma função ou um AWS serviço no Amazon ECR. CloudTrail captura as seguintes ações do Amazon ECR como eventos:

- Todas as chamadas de API, incluindo chamadas do console do Amazon ECR
- Todas as ações tomadas devido às configurações de criptografia em seus repositórios
- Todas as ações tomadas devido às regras de política de ciclo de vida, incluindo ações bem-sucedidas e malsucedidas

> **Important**
>
> Devido às limitações de tamanho de CloudTrail eventos individuais, para ações de política de ciclo de vida em que 10 ou mais imagens expiram, o Amazon ECR envia vários eventos para. CloudTrail Além disso, o Amazon ECR inclui no máximo 100 etiquetas por imagem.

Quando uma trilha é criada, você pode habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para o Amazon ECR. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos. Usando essas informações, é possível determinar a solicitação feita ao Amazon ECR, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Para obter mais informações, consulte o Manual do usuário do AWS CloudTrail.

Informações do Amazon ECR em CloudTrail

CloudTrail é ativado em sua AWS conta quando você cria a conta. Quando a atividade ocorre no Amazon ECR, essa atividade é registrada em um CloudTrail evento junto com outros eventos de AWS serviço no histórico de eventos. Você pode visualizar, pesquisar e baixar eventos recentes em sua AWS conta. Para obter mais informações, consulte Visualização de eventos com histórico de CloudTrail eventos.

Para um registro contínuo de eventos em sua AWS conta, incluindo eventos para o Amazon ECR, crie uma trilha. Uma trilha permite CloudTrail entregar arquivos de log para um bucket do Amazon S3. Ao criar uma trilha no console, você pode aplicá-la a uma única região ou a todas as regiões. A trilha registra eventos na AWS partição e entrega os arquivos de log ao bucket do Amazon S3 que você especificar. Além disso, você pode configurar outros AWS serviços para analisar e agir com base nos dados de eventos coletados nos CloudTrail registros. Para obter mais informações, consulte:

- Criando uma trilha para sua AWS conta
- AWS integrações de serviços com registros CloudTrail
- Configurando notificações do Amazon SNS para CloudTrail
- Recebendo arquivos de CloudTrail log de várias regiões e Recebendo arquivos de CloudTrail log de várias contas

Todas as ações da API do Amazon ECR são registradas CloudTrail e documentadas na Amazon Elastic Container Registry API Reference. Quando você executa tarefas comuns, as seções são geradas nos arquivos de CloudTrail log para cada ação da API que faz parte dessa tarefa. Por exemplo, quando você cria um repositório,`GetAuthorizationToken`, `CreateRepository` e `SetRepositoryPolicy` seções são geradas nos arquivos de CloudTrail log. Quando você envia uma imagem para um repositório, são geradas as seções `InitiateLayerUpload`, `UploadLayerPart`, `CompleteLayerUpload` e `PutImage`. Quando você extrai uma imagem, são geradas as seções `GetDownloadUrlForLayer` e `BatchGetImage`. Para ver exemplos dessas tarefas comuns, consulte CloudTrail exemplos de entrada de registro.

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 do

- Se a solicitação foi feita com credenciais de segurança temporárias de um perfil ou de um usuário federado
- Se a solicitação foi feita por outro AWS serviço

Para obter mais informações, consulte o CloudTrail`userIdentity`Elemento.

Noções básicas sobre entradas do arquivo de log do Amazon ECR

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3 que você especificar. CloudTrail os arquivos de log contêm uma ou mais entradas de log. Um evento representa uma única solicitação de qualquer fonte e inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros da solicitação e outras informações. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das chamadas públicas de API, portanto, eles não aparecem em nenhuma ordem específica.

CloudTrail exemplos de entrada de registro

Veja a seguir exemplos de entrada de CloudTrail registro para algumas tarefas comuns do Amazon ECR.

> **Note**
>
> Estes exemplos foram formatados para obter melhor legibilidade. Em um arquivo de CloudTrail log, todas as entradas e eventos são concatenados em uma única linha. Além disso, este exemplo foi limitado a uma única entrada do Amazon ECR. Em um arquivo de CloudTrail log real, você vê entradas e eventos de vários AWS serviços.

Tópicos

- Exemplo: criar ação de repositório
- Exemplo: ação de AWS KMS CreateGrant API ao criar um repositório Amazon ECR
- Exemplo: ação de envio de imagem
- Exemplo: ação de extração de imagem
- Exemplo: ação da política de ciclo de vida da imagem

Exemplo: criar ação de repositório

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a `CreateRepository` ação.


```
{
    "eventVersion": "1.04",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE:account_name",
        "arn": "arn:aws:sts::123456789012:user/Mary_Major",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-07-11T21:54:07Z"
            },
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::123456789012:role/Admin",
                "accountId": "123456789012",
                "userName": "Admin"
            }
        }
    },
    "eventTime": "2018-07-11T22:17:43Z",
    "eventSource": "ecr.amazonaws.com",
    "eventName": "CreateRepository",
    "awsRegion": "us-east-2",
    "sourceIPAddress": "203.0.113.12",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
        "repositoryName": "testrepo"
    },
    "responseElements": {
        "repository": {
            "repositoryArn": "arn:aws:ecr:us-east-2:123456789012:repository/testrepo",
            "repositoryName": "testrepo",
            "repositoryUri": "123456789012.dkr.ecr.us-east-2.amazonaws.com/testrepo",
            "createdAt": "Jul 11, 2018 10:17:44 PM",
            "registryId": "123456789012"
        }
```

```
    },
    "requestID": "cb8c167e-EXAMPLE",
    "eventID": "e3c6f4ce-EXAMPLE",
    "resources": [
        {
            "ARN": "arn:aws:ecr:us-east-2:123456789012:repository/testrepo",
            "accountId": "123456789012"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```

Exemplo: ação de AWS KMS CreateGrant API ao criar um repositório Amazon ECR

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a AWS KMS CreateGrant ação ao criar um repositório Amazon ECR com a criptografia KMS ativada. Para cada repositório criado com a criptografia KMS ativada, você deverá ver duas entradas de CreateGrant registro. CloudTrail

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAIEP6W46J43IG7LXAQ",
        "arn": "arn:aws:iam::123456789012:user/Mary_Major",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "Mary_Major",
        "sessionContext": {
            "sessionIssuer": {

            },
            "webIdFederationData": {

            },
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2020-06-10T19:22:10Z"
            }
        },
        "invokedBy": "AWS Internal"
    },
```

```
    "eventTime": "2020-06-10T19:22:10Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateGrant",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "203.0.113.12",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
        "keyId": "4b55e5bf-39c8-41ad-b589-18464af7758a",
        "granteePrincipal": "ecr.us-west-2.amazonaws.com",
        "operations": [
            "GenerateDataKey",
            "Decrypt"
        ],
        "retiringPrincipal": "ecr.us-west-2.amazonaws.com",
        "constraints": {
            "encryptionContextSubset": {
                "aws:ecr:arn": "arn:aws:ecr:us-west-2:123456789012:repository/testrepo"
            }
        }
    },
    "responseElements": {
        "grantId": "3636af9adfee1accb67b83941087dcd45e7fadc4e74ff0103bb338422b5055f3"
    },
    "requestID": "047b7dea-b56b-4013-87e9-a089f0f6602b",
    "eventID": "af4c9573-c56a-4886-baca-a77526544469",
    "readOnly": false,
    "resources": [
        {
            "accountId": "123456789012",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:123456789012:key/4b55e5bf-39c8-41ad-
b589-18464af7758a"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```


Exemplo: ação de envio de imagem

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra um push de imagem que usa a `PutImage` ação.

> **Note**
>
> Ao enviar uma imagem, você também verá `InitiateLayerUploadUploadLayerPart`, e `CompleteLayerUpload` referências nos CloudTrail registros.


```
{
    "eventVersion": "1.04",
    "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE:account_name",
    "arn": "arn:aws:sts::123456789012:user/Mary_Major",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
  "userName": "Mary_Major",
  "sessionContext": {
   "attributes": {
    "mfaAuthenticated": "false",
    "creationDate": "2019-04-15T16:42:14Z"
   }
  }
 },
 "eventTime": "2019-04-15T16:45:00Z",
 "eventSource": "ecr.amazonaws.com",
 "eventName": "PutImage",
 "awsRegion": "us-east-2",
 "sourceIPAddress": "203.0.113.12",
 "userAgent": "console.amazonaws.com",
 "requestParameters": {
  "repositoryName": "testrepo",
  "imageTag": "latest",
  "registryId": "123456789012",
  "imageManifest": "{\n   \"schemaVersion\": 2,\n   \"mediaType\": \"application/
vnd.docker.distribution.manifest.v2+json\",\n   \"config\": {\n      \"mediaType\":
 \"application/vnd.docker.container.image.v1+json\",\n      \"size\": 5543,\n
 \"digest\": \"sha256:000b9b805af1cdb60628898c9f411996301a1c13afd3dbef1d8a16ac6dbf503a
\"\n   },\n   \"layers\": [\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 43252507,\n
 \"digest\": \"sha256:3b37166ec61459e76e33282dda08f2a9cd698ca7e3d6bc44e6a6e7580cdeff8e
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 846,\n         \"digest
\": \"sha256:504facff238fde83f1ca8f9f54520b4219c5b8f80be9616ddc52d31448a044bd
```

```
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 615,\n         \"digest
\": \"sha256:ebbcacd28e101968415b0c812b2d2dc60f969e36b0b08c073bf796e12b1bb449\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 850,\n         \"digest
\": \"sha256:c7fb3351ecad291a88b92b600037e2435c84a347683d540042086fe72c902b8a
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 168,\n         \"digest\":
 \"sha256:2e3debadcbf7e542e2aefbce1b64a358b1931fb403b3e4aeca27cb4d809d56c2\"\n      },
\n      {\n         \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip
\",\n         \"size\": 37720774,\n         \"digest\":
 \"sha256:f8c9f51ad524d8ae9bf4db69cd3e720ba92373ec265f5c390ffb21bb0c277941\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 30432107,\n
 \"digest\": \"sha256:813a50b13f61cf1f8d25f19fa96ad3aa5b552896c83e86ce413b48b091d7f01b
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 197,\n         \"digest
\": \"sha256:7ab043301a6187ea3293d80b30ba06c7bf1a0c3cd4c43d10353b31bc0cecfe7d
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 154,\n         \"digest
\": \"sha256:67012cca8f31dc3b8ee2305e7762fee20c250513effdedb38a1c37784a5a2e71\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 176,\n         \"digest
\": \"sha256:3bc892145603fffc9b1c97c94e2985b4cb19ca508750b15845a5d97becbd1a0e
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 183,\n         \"digest
\": \"sha256:6f1c79518f18251d35977e7e46bfa6c6b9cf50df2a79d4194941d95c54258d18\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 212,\n         \"digest
\": \"sha256:b7bcfbc2e2888afebede4dd1cd5eebf029bb6315feeaf0b56e425e11a50afe42\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 212,\n         \"digest\":
 \"sha256:2b220f8b0f32b7c2ed8eaafe1c802633bbd94849b9ab73926f0ba46cdae91629\"\n      }\n
   ]\n}"
 },
 "responseElements": {
  "image": {
   "repositoryName": "testrepo",
   "imageManifest": "{\n   \"schemaVersion\": 2,\n   \"mediaType\": \"application/
vnd.docker.distribution.manifest.v2+json\",\n   \"config\": {\n      \"mediaType\":
 \"application/vnd.docker.container.image.v1+json\",\n      \"size\": 5543,\n
 \"digest\": \"sha256:000b9b805af1cdb60628898c9f411996301a1c13afd3dbef1d8a16ac6dbf503a
\"\n   },\n   \"layers\": [\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 43252507,\n
```

```
 \"digest\": \"sha256:3b37166ec61459e76e33282dda08f2a9cd698ca7e3d6bc44e6a6e7580cdeff8e
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 846,\n         \"digest
\": \"sha256:504facff238fde83f1ca8f9f54520b4219c5b8f80be9616ddc52d31448a044bd
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 615,\n         \"digest
\": \"sha256:ebbcacd28e101968415b0c812b2d2dc60f969e36b0b08c073bf796e12b1bb449\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 850,\n         \"digest
\": \"sha256:c7fb3351ecad291a88b92b600037e2435c84a347683d540042086fe72c902b8a
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 168,\n         \"digest\":
 \"sha256:2e3debadcbf7e542e2aefbce1b64a358b1931fb403b3e4aeca27cb4d809d56c2\"\n      },
\n      {\n         \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip
\",\n         \"size\": 37720774,\n         \"digest\":
 \"sha256:f8c9f51ad524d8ae9bf4db69cd3e720ba92373ec265f5c390ffb21bb0c277941\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 30432107,\n
 \"digest\": \"sha256:813a50b13f61cf1f8d25f19fa96ad3aa5b552896c83e86ce413b48b091d7f01b
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 197,\n         \"digest
\": \"sha256:7ab043301a6187ea3293d80b30ba06c7bf1a0c3cd4c43d10353b31bc0cecfe7d
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 154,\n         \"digest
\": \"sha256:67012cca8f31dc3b8ee2305e7762fee20c250513effdedb38a1c37784a5a2e71\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 176,\n         \"digest
\": \"sha256:3bc892145603fffc9b1c97c94e2985b4cb19ca508750b15845a5d97becbd1a0e
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 183,\n         \"digest
\": \"sha256:6f1c79518f18251d35977e7e46bfa6c6b9cf50df2a79d4194941d95c54258d18\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 212,\n         \"digest
\": \"sha256:b7bcfbc2e2888afebede4dd1cd5eebf029bb6315feeaf0b56e425e11a50afe42\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 212,\n         \"digest\":
 \"sha256:2b220f8b0f32b7c2ed8eaafe1c802633bbd94849b9ab73926f0ba46cdae91629\"\n      }\n
   ]\n}",
   "registryId": "123456789012",
   "imageId": {
    "imageDigest":
 "sha256:98c8b060c21d9adbb6b8c41b916e95e6307102786973ab93a41e8b86d1fc6d3e",
    "imageTag": "latest"
   }
```

```
  }
 },
 "requestID": "cf044b7d-5f9d-11e9-9b2a-95983139cc57",
 "eventID": "2bfd4ee2-2178-4a82-a27d-b12939923f0f",
 "resources": [{
  "ARN": "arn:aws:ecr:us-east-2:123456789012:repository/testrepo",
  "accountId": "123456789012"
 }],
 "eventType": "AwsApiCall",
 "recipientAccountId": "123456789012"
}
```

Exemplo: ação de extração de imagem

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra uma extração de imagem que usa a `BatchGetImage` ação.

> **Note**
>
> Ao extrair uma imagem, se você ainda não tiver a imagem localmente, você também verá `GetDownloadUrlForLayer` referências nos CloudTrail registros.

```
{
    "eventVersion": "1.04",
    "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE:account_name",
    "arn": "arn:aws:sts::123456789012:user/Mary_Major",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
  "userName": "Mary_Major",
  "sessionContext": {
   "attributes": {
    "mfaAuthenticated": "false",
    "creationDate": "2019-04-15T16:42:14Z"
   }
  }
 },
 "eventTime": "2019-04-15T17:23:20Z",
 "eventSource": "ecr.amazonaws.com",
 "eventName": "BatchGetImage",
```

```
 "awsRegion": "us-east-2",
 "sourceIPAddress": "203.0.113.12",
 "userAgent": "console.amazonaws.com",
 "requestParameters": {
  "imageIds": [{
   "imageTag": "latest"
  }],
  "acceptedMediaTypes": [
   "application/json",
   "application/vnd.oci.image.manifest.v1+json",
   "application/vnd.oci.image.index.v1+json",
   "application/vnd.docker.distribution.manifest.v2+json",
   "application/vnd.docker.distribution.manifest.list.v2+json",
   "application/vnd.docker.distribution.manifest.v1+prettyjws"
  ],
  "repositoryName": "testrepo",
  "registryId": "123456789012"
 },
 "responseElements": null,
 "requestID": "2a1b97ee-5fa3-11e9-a8cd-cd2391aeda93",
 "eventID": "c84f5880-c2f9-4585-9757-28fa5c1065df",
 "resources": [{
  "ARN": "arn:aws:ecr:us-east-2:123456789012:repository/testrepo",
  "accountId": "123456789012"
 }],
 "eventType": "AwsApiCall",
 "recipientAccountId": "123456789012"
}
```


Exemplo: ação da política de ciclo de vida da imagem

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra quando uma imagem expira devido a uma regra de política de ciclo de vida. Esse tipo de evento pode ser localizado filtrando `PolicyExecutionEvent` para o campo de nome do evento.

> **Important**
>
> Devido às limitações de tamanho de CloudTrail eventos individuais, para ações de política de ciclo de vida em que 10 ou mais imagens expiram, o Amazon ECR envia vários eventos para. CloudTrail Além disso, o Amazon ECR inclui no máximo 100 etiquetas por imagem.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "accountId": "123456789012",
        "invokedBy": "AWS Internal"
    },
    "eventTime": "2020-03-12T20:22:12Z",
    "eventSource": "ecr.amazonaws.com",
    "eventName": "PolicyExecutionEvent",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": null,
    "responseElements": null,
    "eventID": "9354dd7f-9aac-4e9d-956d-12561a4923aa",
    "readOnly": true,
    "resources": [
        {
            "ARN": "arn:aws:ecr:us-west-2:123456789012:repository/testrepo",
            "accountId": "123456789012",
            "type": "AWS::ECR::Repository"
        }
    ],
    "eventType": "AwsServiceEvent",
    "recipientAccountId": "123456789012",
    "serviceEventDetails": {
        "repositoryName": "testrepo",
        "lifecycleEventPolicy": {
            "lifecycleEventRules": [
                {
                    "rulePriority": 1,
                    "description": "remove all images > 2",
                    "lifecycleEventSelection": {
                        "tagStatus": "Any",
                        "tagPrefixList": [],
                        "countType": "Image count more than",
                        "countNumber": 2
                    },
                    "action": "expire"
                }
            ],
            "lastEvaluatedAt": 0,
            "policyVersion": 1,
```

```
            "policyId": "ceb86829-58e7-9498-920c-aa042e33037b"
        },
        "lifecycleEventImageActions": [
            {
                "lifecycleEventImage": {
                    "digest":
 "sha256:ddba4d27a7ffc3f86dd6c2f92041af252a1f23a8e742c90e6e1297bfa1bc0c45",
                    "tagStatus": "Tagged",
                    "tagList": [
                        "alpine"
                    ],
                    "pushedAt": 1584042813000
                },
                "rulePriority": 1
            },
            {
                "lifecycleEventImage": {
                    "digest":
 "sha256:6ab380c5a5acf71c1b6660d645d2cd79cc8ce91b38e0352cbf9561e050427baf",
                    "tagStatus": "Tagged",
                    "tagList": [
                        "centos"
                    ],
                    "pushedAt": 1584042842000
                },
                "rulePriority": 1
            }
        ]
    }
}
```

Usando o Amazon ECR com um SDK AWS

AWS kits de desenvolvimento de software (SDKs) estão disponíveis para muitas linguagens de programação populares. Cada SDK fornece uma API, exemplos de código e documentação que facilitam a criação de aplicações em seu idioma preferido pelos desenvolvedores.

Documentação do SDK	Exemplos de código
AWS SDK for C++	AWS SDK for C++ exemplos de código
AWS CLI	AWS CLI exemplos de código
AWS SDK for Go	AWS SDK for Go exemplos de código
AWS SDK for Java	AWS SDK for Java exemplos de código
AWS SDK for JavaScript	AWS SDK for JavaScript exemplos de código
AWS SDK para Kotlin	AWS SDK para Kotlin exemplos de código
AWS SDK for .NET	AWS SDK for .NET exemplos de código
AWS SDK for PHP	AWS SDK for PHP exemplos de código
AWS Tools for PowerShell	Ferramentas para exemplos PowerShell de código
AWS SDK for Python (Boto3)	AWS SDK for Python (Boto3) exemplos de código
AWS SDK for Ruby	AWS SDK for Ruby exemplos de código
AWS SDK para Rust	AWS SDK para Rust exemplos de código
SDK da AWS para SAP ABAP	SDK da AWS para SAP ABAP exemplos de código
AWS SDK for Swift	AWS SDK for Swift exemplos de código

Exemplo de disponibilidade

Você não consegue encontrar o que precisa? Solicite um código de exemplo no link Fornecer feedback na parte inferior desta página.

Exemplos de código para Amazon ECR usando AWS SDKs

Os exemplos de código a seguir mostram como usar o Amazon ECR com um kit de desenvolvimento AWS de software (SDK).

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar funções de serviço específicas, é possível ver as ações contextualizadas em seus devidos cenários e exemplos entre serviços.

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUsando o Amazon ECR com um SDK AWS. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Exemplos de código

- Ações para o Amazon ECR usando AWS SDKs
  - Use DescribeRepositories com um AWS SDK ou CLI
  - Use ListImages com um AWS SDK ou CLI

Ações para o Amazon ECR usando AWS SDKs

Os exemplos de código a seguir demonstram como realizar ações individuais do Amazon ECR com AWS SDKs. Esses trechos chamam a API Amazon ECR e são trechos de código de programas maiores que devem ser executados em contexto. Cada exemplo inclui um link para GitHub, onde você pode encontrar instruções para configurar e executar o código.

Os exemplos a seguir incluem apenas as ações mais utilizadas. Para obter uma lista completa, consulte a referência da API do Amazon Elastic Container Registry (Amazon ECR).

Exemplos

- Use DescribeRepositories com um AWS SDK ou CLI
- Use ListImages com um AWS SDK ou CLI

Use `DescribeRepositories` com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `DescribeRepositories`.

CLI

AWS CLI

Como descrever os repositórios em um registro

Este exemplo descreve os repositórios no registro padrão de uma conta.

Comando:

```
aws ecr describe-repositories
```

Saída:

```
{
    "repositories": [
        {
            "registryId": "012345678910",
            "repositoryName": "ubuntu",
            "repositoryArn": "arn:aws:ecr:us-west-2:012345678910:repository/
ubuntu"
        },
        {
            "registryId": "012345678910",
            "repositoryName": "test",
            "repositoryArn": "arn:aws:ecr:us-west-2:012345678910:repository/test"
        }
    ]
}
```

- Para obter detalhes da API, consulte DescribeRepositoriesem Referência de AWS CLI Comandos.

Rust

SDK para Rust

> **Note**
>
> Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no AWS Code Examples Repository.

```
async fn show_repos(client: &aws_sdk_ecr::Client) -> Result<(),
 aws_sdk_ecr::Error> {
    let rsp = client.describe_repositories().send().await?;

    let repos = rsp.repositories();

    println!("Found {} repositories:", repos.len());

    for repo in repos {
        println!("  ARN:  {}", repo.repository_arn().unwrap());
        println!("  Name: {}", repo.repository_name().unwrap());
    }

    Ok(())
}
```

- Para obter detalhes da API, consulte a DescribeRepositoriesreferência da API AWS SDK for Rust.

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUsando o Amazon ECR com um SDK AWS. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`ListImages`** com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `ListImages`.

CLI

AWS CLI

Como listar as imagens em um repositório

O exemplo de `list-images` a seguir exibe uma lista das imagens presentes no repositório `cluster-autoscaler`.

```
aws ecr list-images \
    --repository-name cluster-autoscaler
```

Saída:

```
{
    "imageIds": [
        {
            "imageDigest":
 "sha256:99c6fb4377e9a420a1eb3b410a951c9f464eff3b7dbc76c65e434e39b94b6570",
            "imageTag": "v1.13.8"
        },
        {
            "imageDigest":
 "sha256:99c6fb4377e9a420a1eb3b410a951c9f464eff3b7dbc76c65e434e39b94b6570",
            "imageTag": "v1.13.7"
        },
        {
            "imageDigest":
 "sha256:4a1c6567c38904384ebc64e35b7eeddd8451110c299e3368d2210066487d97e5",
            "imageTag": "v1.13.6"
        }
    ]
}
```

- Para obter detalhes da API, consulte ListImagesem Referência de AWS CLI Comandos.

Rust

SDK para Rust

> **Note**
>
> Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no AWS Code Examples Repository.

```
async fn show_images(
    client: &aws_sdk_ecr::Client,
    repository: &str,
) -> Result<(), aws_sdk_ecr::Error> {
    let rsp = client
        .list_images()
        .repository_name(repository)
        .send()
        .await?;
```

```
    let images = rsp.image_ids();

    println!("found {} images", images.len());

    for image in images {
        println!(
            "image: {}:{}",
            image.image_tag().unwrap(),
            image.image_digest().unwrap()
        );
    }

    Ok(())
}
```

- Para obter detalhes da API, consulte a ListImagesreferência da API AWS SDK for Rust.

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUsando o Amazon ECR com um SDK AWS. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Cotas de serviço do Amazon ECR

A tabela a seguir fornece as cotas de serviço padrão do Amazon Elastic Container Registry (Amazon ECR).

Nome	Padrão	Ajustá	Descrição
Filtros por regra em uma configuração de replicação	Cada região com suporte: 100	Não	O número máximo de filtros por regra em uma configuração de replicaçã o.
Imagem por repositório	Cada região compatível: 10.000	Sim	O número máximo de imagens por repositório.
Partes da camada	Cada região compatível: 4.200	Não	O número máximo de partes da camada. Isso é aplicável somente se você estiver usando as ações da API do Amazon ECR diretamente para iniciar multipart uploads para operações de envio de imagem.
Duração da política de ciclo de vida	Cada região compatível: 30.720	Não	O número máximo de caracteres em uma política de ciclo de vida.
Tamanho máximo da parte da camada	Cada região com suporte: 10	Não	O tamanho máximo (MiB) de uma parte da camada. Isso é aplicável somente se você estiver usando as ações da API do Amazon ECR diretamente para iniciar multipart uploads

Nome	Padrão	Ajustá	Descrição
			para operações de envio de imagem.
Tamanho máximo da camada	Cada região compatível: 52.000	Não	O tamanho máximo (MiB) de uma camada.
Tamanho mínimo da parte da camada	Cada região compatível: 5	Não	O tamanho mínimo (MiB) de uma parte da camada. Isso é aplicável somente se você estiver usando as ações da API do Amazon ECR diretamente para iniciar multipart uploads para operações de envio de imagem.
Regras de cache de pull-through por registro	Cada região compatível: 50	Não	O número máximo de regras de cache de pull-through.
Taxa de solicitações de BatchChecLayerAvailability	Cada região compatível: 1.000 por segundo	Sim	O número máximo de solicitações de BatchCheckLayerAvailability que podem ser feitas por segundo na região atual. Quando uma imagem é enviada para um repositório, cada camada da imagem é conferida para verificar se foi feito upload dela anteriormente. Se o upload já tiver sido feito, a camada da imagem será ignorada.

Nome	Padrão	Ajustá	Descrição
Taxa de solicitações de BatchGetImage	Cada região compatível: 2.000 por segundo	Sim	O número máximo de solicitações de BatchGetImage que podem ser feitas por segundo na região atual. Quando uma imagem é extraída, a API BatchGetImage é chamada uma vez para recuperar o manifesto da imagem. Se você solicitar um aumento de cota para essa API, revise seu uso de GetDownloadUrlForLayer também.
Taxa de solicitações de CompleteLayerUpload	Cada região compatível: 100 por segundo	Sim	O número máximo de solicitações de CompleteLayerUpload que podem ser feitas por segundo na região atual. Quando uma imagem é enviada, a API CompleteLayerUpload é chamada uma vez por cada nova camada de imagem para verificar se o upload foi concluído.
Taxa de solicitações de GetAuthorizationToken	Cada região compatível: 500 por segundo	Sim	O número máximo de solicitações de GetAuthorizationToken que podem ser feitas por segundo na região atual.

Nome	Padrão	Ajustá	Descrição
Taxa de solicitações de GetDownloadUrlForLayer	Cada região compatível: 3.000 por segundo	Sim	O número máximo de solicitações de GetDownloadUrlForLayer que podem ser feitas por segundo na região atual. Quando uma imagem é extraída, a API GetDownloadUrlForLayer é chamada uma vez por camada de imagem que ainda não está armazenada em cache. Se você solicitar um aumento de cota para essa API, revise seu uso de BatchGetImage também.
Taxa de solicitações de InitiateLayerUpload	Cada região compatível: 100 por segundo	Sim	O número máximo de solicitações de InitiateLayerUpload que podem ser feitas por segundo na região atual. Quando uma imagem é enviada, a API InitiateLayerUpload é chamada uma vez por camada de imagem da qual ainda não foi feito upload. A ação da API BatchCheckLayerAvailability é que determina se foi feito ou não upload de uma camada de imagem.

Nome	Padrão	Ajustá	Descrição
Taxa de solicitações de PutImage	Cada região compatível: 10 por segundo	Sim	O número máximo de solicitações de PutImage que podem ser feitas por segundo na região atual. Quando uma imagem é enviada e é feito upload de todas as novas camadas da imagem, a API PutImage é chamada uma vez para criar ou atualizar o manifesto da imagem e as tags associadas à imagem.
Taxa de solicitações de UploadLayerPart	Cada região compatível: 500 por segundo	Sim	O número máximo de solicitações de UploadLayerPart que podem ser feitas por segundo na região atual. Quando uma imagem é enviada, é feito upload de cada nova camada da imagem em partes, e a API UploadLayerPart é chamada uma vez por cada nova parte da camada da imagem.
Taxa de verificações de imagens	Cada região compatível: 1	Não	O número máximo de verificações de imagem por imagem, a cada 24 horas.

Nome	Padrão	Ajustá	Descrição
Repositórios registrados	Cada região compatível: 10.000	Sim	O número máximo de repositórios que você pode criar nesta conta na região atual.
Regras por política de ciclo de vida	Cada região compatível: 50	Não	O número máximo de regras em uma política de ciclo de vida
Regras por configuração de replicação	Cada região com suporte: 10	Não	O número máximo de regras em uma configura ção de replicação.
Tags por imagem	Cada região compatível: 1.000	Não	O número máximo de tags por imagem.
Destinos exclusivos em todas as regras em uma configuração de replicação	Cada região compatível: 25	Não	O número máximo de destinos exclusivos em todas as regras em uma configuração de replicaçã o.

Gerenciamento de cotas de serviço do Amazon ECR no AWS Management Console

O Amazon ECR foi integrado ao Service Quotas, um serviço da AWS que permite visualizar e gerenciar as cotas em um local central. Para obter mais informações, consulte O que é são cotas de serviço? no Manual do usuário do Service Quotas.

O Service Quotas facilita a pesquisa do valor de todas as cotas de serviço do Amazon ECR.

Consultar as cotas de serviço do Amazon ECR (AWS Management Console)

1. Abra o console Service Quotas em https://console.aws.amazon.com/servicequotas/.
2. No painel de navegação, selecione AWS serviços.

3. Na lista de serviços da AWS, procure e selecione Amazon Elastic Container Registry (Amazon ECR).

   Na lista Service quotas é possível ver o nome da service quota, o valor aplicado (se estiver disponível), AWS a cota padrão e se o valor da cota é ajustável.
4. Para visualizar informações adicionais sobre uma Service Quota, como a descrição, escolha o nome da cota.

Para solicitar um aumento de cota, consulte Solicitação de um aumento de cota no Manual do usuário do Service Quotas.

Criação de um alarme do CloudWatch para monitorar as métricas de uso da API

O Amazon ECR fornece métricas de uso do CloudWatch que correspondem às cotas de serviço da AWS para cada uma das APIs envolvidas com as ações de autenticação de registro, envio de imagem e extração de imagem. No console do Service Quotas, é possível visualizar o uso em um gráfico e configurar alarmes que alertarão você quando o uso se aproximar de uma cota de serviço. Para obter mais informações, consulte Métricas de uso do Amazon ECR.

Use as etapas a seguir para criar um alarme do CloudWatch baseado em uma das métricas de uso da API do Amazon ECR.

Para criar um alarme baseado nas cotas de uso do Amazon ECR (AWS Management Console)

1. Abra o console Service Quotas em https://console.aws.amazon.com/servicequotas/.
2. No painel de navegação, selecione AWS serviços.
3. Na lista de serviços da AWS, procure e selecione Amazon Elastic Container Registry (Amazon ECR).
4. Na lista de cotas de serviço, selecione a cota de uso do Amazon ECR para a qual você deseja criar um alarme.
5. Na seção de alarmes do Amazon CloudWatch Events, escolha Create da (Criar).
6. Em Alarm threshold (Limitação do alarme), escolha a porcentagem do valor da cota aplicada que você deseja definir como o valor do alarme.
7. Em Alarm name (Nome do alarme), insira um nome para o alarme e selecione Create (Criar).

Solução de problemas do Amazon ECR

Este capítulo ajuda você a encontrar informações de diagnóstico para o Amazon ECR e fornece etapas de solução de problemas comuns e mensagens de erro.

Tópicos

Solução de problemas e comandos do Docker ao usar o Amazon ECR

Em alguns casos, executar um comando do Docker no Amazon ECR pode resultar em uma mensagem de erro. Algumas mensagens de erro comuns e possíveis soluções são explicadas abaixo.

Tópicos

Os registros do Docker não contêm mensagens de erro esperadas

Para começar a depurar qualquer problema relacionado ao Docker, comece ativando a saída de depuração do Docker no daemon do Docker em execução nas suas instâncias hospedeiras. Se você estiver usando imagens extraídas do Amazon ECR em instâncias de contêiner do Amazon ECS, consulte Configuração da saída detalhada do daemon do Docker no Guia do desenvolvedor do Amazon Elastic Container Service.

Erro: "Filesystem Verification Failed (Falha na verificação do sistema de arquivos)" ou "404: Image Not Found (Imagem não encontrada)" ao extrair uma imagem de um repositório do Amazon ECR

Você pode receber o erro `Filesystem verification failed` ao usar o comando docker pull para extrair uma imagem de um repositório do Amazon ECR com o Docker 1.9 ou versões posteriores. Ou o erro `404: Image not found` ao utilizar versões do Docker anteriores à 1.9.

Alguns motivos possíveis e suas explicações são mostrados abaixo.

O disco local está cheio

Se o disco local em que você está executando docker pull estiver cheio, o hash SHA-1 calculado no arquivo local pode ser diferente do calculado pelo Amazon ECR. Verifique se o disco local tem espaço livre suficiente para armazenar a imagem de Docker que você está extraindo. Você também pode excluir imagens antigas para dar espaço às novas. Use o comando docker images para ver uma lista com todas as imagens de Docker obtidas por download localmente, bem como os tamanhos delas.

O cliente não consegue se conectar ao repositório remoto devido a um erro de rede

As chamadas feitas para um repositório do Amazon ECR exigem uma conexão com a Internet. Verifique suas configurações de rede e se outros aplicativos e ferramentas podem acessar recursos na Internet. Se estiver executando docker pull em uma instância do Amazon EC2 em uma sub-rede privada, verifique se a sub-rede tem uma rota para a Internet. Use um servidor de tradução de endereço de rede (NAT) ou um gateway NAT gerenciado.

Atualmente, as chamadas feitas para um repositório do Amazon ECR também exigem acesso de rede por firewall corporativo para o Amazon Simple Storage Service (Amazon S3). Se sua organização usa software de firewall ou um dispositivo NAT que permite endpoints de serviço, verifique se os endpoints de serviço do Amazon S3 para a sua região atual são permitidos.

Se você usa o Docker atrás de um proxy HTTP, pode configurá-lo com as configurações de proxy apropriadas. Para obter mais informações, consulte proxy HTTP na documentação do Docker.

Erro: "Filesystem Layer Verification Failed (Falha na verificação da camada do sistema de arquivos)" ao extrair imagens do Amazon ECR

Você pode receber o erro `image image-name not found` ao extrair imagens com o comando docker pull. Se você inspecionar os logs do Docker, poderá ver um erro como este:

```
filesystem layer verification failed for digest sha256:2b96f...
```

Esse erro indica que uma ou mais das camadas da sua imagem não foram baixadas. Alguns motivos possíveis e suas explicações são mostrados abaixo.

Você está usando uma versão antiga do Docker

Esse erro pode ocorrer em alguns casos, quando uma versão do Docker anterior à 1.10 é usada. Atualize o cliente do Docker para a versão 1.10 ou posterior.

O cliente encontrou um erro de rede ou de disco

Um disco cheio ou um problema de rede pode impedir que uma ou mais camadas sejam baixadas, como já falamos sobre a mensagem `Filesystem verification failed`. Siga as recomendações acima para que seu sistema de arquivos não fique cheio e para verificar se você habilitou o acesso ao Amazon S3 de dentro da sua rede.

Erros 403 de HTTP ou o erro "no basic auth credentials (não há credenciais de autenticação básica)" ao enviar ao repositório

Algumas vezes, você poderá receber um erro `HTTP 403 (Forbidden)` ou a mensagem de erro no `basic auth credentials` dos comandos docker push ou docker pull, mesmo que tenha feito a autenticação no Docker com o comando aws ecr get-login-password. Veja a seguir algumas causas conhecidas desse problema:

Você fez a autenticação em outra região

As solicitações de autenticação são vinculadas a regiões específicas e não podem ser usadas entre regiões. Por exemplo, se você obtiver um token de autorização de Oeste dos EUA (Oregon), não poderá usá-lo para autenticação nos seus repositórios em Leste dos EUA (Norte da Virgínia). Para resolver o problema, verifique se você recuperou um token de autenticação da mesma região na qual o repositório existe. Para ter mais informações, consulte the section called “Autenticação de registro”.

Você realizou uma autenticação para enviar por push para um repositório ao qual não tem permissões

Você não tem as permissões necessárias para realizar o envio por push para o repositório. Para ter mais informações, consulte Políticas de repositório privado no Amazon ECR.

Seu token expirou

O período de expiração padrão para tokens de autorização obtidos usando a operação `GetAuthorizationToken` é de 12 horas.

Erro no gerenciador de credenciais`wincred`

Algumas versões do Docker para Windows usam um gerenciador de credenciais chamado `wincred`, que não gerencia corretamente o comando de login do Docker produzido por aws ecr get-login-password (para obter mais informações, consulte https://github.com/docker/docker/issues/22910). Você pode executar o comando de login do Docker gerado. No entanto, quando você tenta enviar ou extrair imagens, esses comandos falham. Para resolver esse erro, remova o esquema `https://` do argumento de registro no comando de login do Docker gerado a partir de aws ecr get-login-password. Um exemplo de comando de login do Docker sem o esquema HTTPS é mostrado abaixo.

```
docker login -u AWS -p <password> <aws_account_id>.dkr.ecr.<region>.amazonaws.com
```

Solução de problemas de mensagens de erro do Amazon ECR

Em alguns casos, uma chamada de API que você iniciou por meio do console do Amazon ECR ou AWS CLI sai com uma mensagem de erro. Algumas mensagens de erro comuns e possíveis soluções são explicadas abaixo.

HTTP 429: Muitas solicitações ou ThrottleException

Você pode receber um `429: Too Many Requests` erro ou um `ThrottleException` erro de uma ou mais ações do Amazon ECR ou chamadas de API. Isso indica que você está chamando um único endpoint no Amazon ECR repetidamente em um intervalo curto e que suas solicitações estão sendo limitadas. A suspensão ocorre quando as chamadas para um único endpoint de um único usuário ultrapassam um determinado limite em um período.

Cada operação de API no Amazon ECR tem uma limitação de taxa associada a ela. Por exemplo, a suspensão da ação `GetAuthorizationToken` é de 20 transações por segundo (TPS), com

permissão para uma intermitência de até 200 TPS. Em cada região, cada conta recebe um bucket que pode armazenar até 200 créditos `GetAuthorizationToken`. Esses créditos são reabastecidos a uma taxa de 20 por segundo. Se seu bucket tem 200 créditos, você pode alcançar 200 transações de API `GetAuthorizationToken` por segundo e sustentar 20 transações por segundo indefinidamente. Para obter mais informações sobre os limites de taxa para as APIs do Amazon ECR, consulte. Cotas de serviço do Amazon ECR

Para gerenciar os erros de controle de utilização, implemente uma função de novas tentativas com backoff incremental no código. Para obter mais informações, consulte o comportamento de novas tentativas no Guia de referência de AWS SDKs e ferramentas. Outra opção é solicitar um aumento do limite de taxa, o que você pode fazer usando o console Service Quotas. Para obter mais informações, consulte Gerenciamento de cotas de serviço do Amazon ECR no AWS Management Console.

HTTP 403: "O usuário [arn] não está autorizado a executar a [operação]"

Você poderá receber o seguinte erro ao tentar realizar uma ação com o Amazon ECR:

```
$ aws ecr get-login-password
A client error (AccessDeniedException) occurred when calling the GetAuthorizationToken
 operation:
    User: arn:aws:iam::account-number:user/username is not authorized to perform:
    ecr:GetAuthorizationToken on resource: *
```

Isso indica que o usuário não tem as permissões para usar o Amazon ECR ou que essas permissões não estão configuradas corretamente. Se você realizar ações especificamente em um repositório do Amazon ECR, verifique se o usuário recebeu permissões para acessá-lo. Para obter mais informações sobre como criar e verificar permissões do Amazon ECR, consulte Gerenciamento de Identidade e Acesso para o Amazon Elastic Container Registry.

HTTP 404: erro "Repository Does Not Exist (O repositório não existe)"

Se você especificar um repositório do Docker Hub que não existe atualmente, o Docker Hub o criará automaticamente. Com o Amazon ECR, novos repositórios devem ser criados explicitamente para que poder serem usados. Isso impede que novos repositórios sejam criados de maneira acidental (por exemplo, devido a erros de digitação) e também garante que uma política de acesso de segurança apropriada seja atribuída explicitamente a todos os novos repositórios. Para obter mais informações sobre como criar repositórios, consulte Repositórios privados do Amazon ECR.

Erro: não é possível realizar um login interativo em um dispositivo não TTY

Se você receber o erro `Cannot perform an interactive login from a non TTY device`, as etapas de solução de problemas a seguir devem ajudar.

- Verifique se você está usando a AWS CLI versão 2 e se não tem uma versão conflitante da AWS CLI versão 1 em seu sistema. Para obter mais informações, consulte Instalar ou atualizar a versão mais recente da AWS CLI.
- Verifique se você configurou seu AWS CLI com credenciais válidas. Para obter mais informações, consulte Instalar ou atualizar a versão mais recente da AWS CLI.
- Verifique se a sintaxe do seu AWS CLI comando está correta.

Histórico do documento

A tabela a seguir descreve as alterações importantes na documentação desde a última versão do Amazon ECR . Também atualizamos a documentação com frequência para abordar os comentários enviados por você.

Alteração	Descrição	Data
Foi adicionada a replicação entre regiões e contas às regiões da China	O Amazon ECR adicionou suporte à região da China para filtrar quais repositórios são replicados.	15 de maio de 2024
Registro de GitLab contêiner adicionado para verificar as regras de cache	O Amazon ECR adicionou suporte para criar regras de cache pull through para o registro de GitLab contêiner es.	8 de maio de 2024
Atualização da política de ciclo de vida do Amazon ECR para adicionar suporte ao uso de curingas	O Amazon ECR adicionou suporte para curingas em uma política de ciclo de vida por meio do uso do parâmetro `tagPatternList` em uma regra de política de ciclo de vida. Para ter mais informações, consulte Automatize a limpeza de imagens usando políticas de ciclo de vida no Amazon ECR.	18 de dezembro de 2023
Modelos de criação de repositório do Amazon ECR	O Amazon ECR adicionou suporte para modelos de criação de repositórios. Para ter mais informações, consulte Modelos para controlar repositórios criados durante uma ação de extração do cache.	15 de novembro de 2023
O cache de pull-thro ugh do Amazon ECR foi adicionado, compatível com registros upstream autenticados	O Amazon ECR adicionou suporte ao uso de registros upstream que exigem autenticação para suas regras de cache pull through. Para ter mais informações, consulte Sincronize um registro upstream com um registro privado do Amazon ECR.	15 de novembro de 2023

Alteração	Descrição	Data
AWSECRPullThroughCache_ServiceRolePolicy: atualização para uma política existente	O Amazon ECR adicionou novas permissões à política `AWSECRPullThroughCache_ServiceRolePolicy` . Essas permissões permitem que o Amazon ECR recupere o conteúdo criptografado de um segredo do Secrets Manager. Isso é necessário ao usar uma regra de cache de pull-through para armazenar em cache imagens de um registro upstream que requer autenticação.	15 de novembro de 2023
Assinatura de imagem do Amazon ECR	Amazon ECR e suporte AWS Signer adicional para criar e enviar assinaturas de imagens de contêineres usando o cliente Notary. Para ter mais informações, consulte Assinatura de uma imagem armazenada em um repositório privado do Amazon ECR.	6 de junho de 2023
Foi adicionado o registro de contêiner do Kubernetes para extrair as regras de cache	O Amazon ECR incluiu suporte à criação de regras de cache de pull-through para o registro de contêiner do Kubernetes. Para ter mais informações, consulte Sincronize um registro upstream com um registro privado do Amazon ECR.	1 de junho de 2023
Suporte à duração da verificação aprimorada do Amazon ECR	O Amazon Inspector adicionou suporte para definir a duração pela qual seus repositórios são monitorados quando a verificação aprimorada está habilitada. Para ter mais informações, consulte Alterando a duração aprimorada da digitalização de imagens no Amazon Inspector.	28 de junho de 2022
O Amazon ECR envia métricas de contagem de pull do repositório para a Amazon CloudWatch	O Amazon ECR envia métricas de contagem de pull do repositório para a Amazon. CloudWatch Para ter mais informações, consulte Métricas do repositório do Amazon ECR.	6 de janeiro de 2022

Alteração	Descrição	Data
Suporte ampliado a replicação	O Amazon ECR adicionou suporte para filtragem de quais repositórios são replicados. Para ter mais informações, consulte Replicação de imagens privadas no Amazon ECR.	21 de setembro de 2021
AWS políticas gerenciadas para o Amazon ECR	O Amazon ECR adicionou a documentação das políticas AWS gerenciadas. Para ter mais informações, consulte AWS políticas gerenciadas para o Amazon Elastic Container Registry.	24 de junho de 2021
Replicação entre regiões e entre contas	O Amazon ECR adicionou suporte à definição de configurações de replicação para seu registro privado. Para ter mais informações, consulte Configurações de registro privado no Amazon ECR.	8 de dezembro de 2020
Suporte a artefatos OCI	O Amazon ECR adicionou suporte ao envio e extração de artefatos Open Container Initiative (OCI). Um novo parâmetro `artifactMediaType` foi adicionado à resposta da API `DescribeImages` para indicar o tipo de artefato. Para ter mais informações, consulte Enviando um gráfico do Helm para um repositório privado do Amazon ECR.	24 de agosto de 2020
Criptografia inativa	O Amazon ECR adicionou suporte à configuração de criptografia para seus repositórios usando a criptografia do lado do servidor com as chaves gerenciadas do cliente armazenada no AWS Key Management Service (AWS KMS). Para ter mais informações, consulte Criptografia em repouso.	29 de julho de 2020

Alteração	Descrição	Data
Imagens multiarquitetura	O Amazon ECR adicionou suporte à criação e ao envio de listas de manifesto do Docker usadas para imagens de multiarquitetura. Para ter mais informações, consulte Enviando uma imagem de várias arquiteturas para um repositório privado do Amazon ECR.	28 de abril de 2020
Métricas de uso do Amazon ECR	O Amazon ECR adicionou métricas CloudWatch de uso que fornecem visibilidade ao uso de recursos da sua conta. Você também pode criar CloudWatch alarmes nos consoles CloudWatch e Service Quotas para receber alertas quando seu uso se aproximar da cota de serviço aplicada. Para ter mais informações, consulte Métricas de uso do Amazon ECR.	28 de fevereiro de 2020
Cotas de serviço do Amazon ECR atualizadas	Atualização das cotas de serviço do Amazon ECR para incluir cotas por API. Para ter mais informações, consulte Cotas de serviço do Amazon ECR.	19 de fevereiro de 2020
Adição do comando `get-login-password`	Adição de suporte para get-login-password, que oferece um método simples e seguro para recuperar um token de autorização. Para ter mais informações, consulte Uso de um token de autorização.	4 de fevereiro de 2020

Alteração	Descrição	Data
Verificação de imagens	Adição de suporte à verificação de imagens, o que ajuda na identificação de vulnerabilidades de software em suas imagens de contêiner. O Amazon ECR usa o banco de dados de vulnerabilidades e exposições comuns (CVEs) do projeto CoreOS Clair de código aberto e fornece uma lista das descobertas da verificação. Para ter mais informações, consulte Digitalize imagens em busca de vulnerabilidades de software no Amazon ECR.	24 de outubro de 2019
Política de VPC endpoint	Adição de suporte à configuração de uma política do IAM nos endpoints da interface da VPC do Amazon ECR. Para ter mais informações, consulte Criar uma política de endpoint para os endpoints da VPC do Amazon ECR.	26 de setembro de 2019
Mutabilidade de tag de imagem	Adição de suporte à configuração de um repositór io para ser imutável a fim de impedir que as tags de imagem sejam substituídas. Para ter mais informações, consulte Impedindo que as tags de imagem sejam sobrescritas no Amazon ECR.	25 de julho de 2019
Endpoints da VPC de interface (AWS PrivateLi nk)	Foi adicionado suporte para configurar endpoints VPC de interface desenvolvidos por. AWS PrivateLink Isso permite criar uma conexão privada entre sua VPC e o Amazon ECR, sem necessidade de acesso pela Internet, por meio de uma instância NAT, de uma conexão VPN ou do AWS Direct Connect. Para ter mais informações, consulte Endpoints VPC da interface Amazon ECR ()AWS PrivateLink.	25 de janeiro de 2019

Alteração	Descrição	Data
Marcação de recursos	O Amazon ECR adicionou suporte à adição de tags de metadados aos seus repositórios. Para ter mais informações, consulte Marcar um repositório privado no Amazon ECR.	18 de dezembro de 2018
Alteração de nome do Amazon ECR	O Amazon Elastic Container Registry foi renomeado (anteriormente, Amazon EC2 Container Registry).	21 de novembro de 2017
Políticas de ciclo de vida	As políticas de ciclo de vida do Amazon ECR permitem que você especifique o gerenciamento do ciclo de vida das imagens em um repositório. Para ter mais informações, consulte Automatize a limpeza de imagens usando políticas de ciclo de vida no Amazon ECR.	11 de outubro de 2017
Suporte do Amazon ECR ao manifesto V2 esquema 2 de imagem do Docker	O Amazon ECR agora suporta o manifesto V2 esquema 2 de imagem do Docker (usado com o Docker versão 1.10 e posteriores). Para ter mais informações, consulte Suporte ao formato de manifesto de imagem de contêiner no Amazon ECR.	27 de janeiro de 2017
Disponibilidade geral do Amazon ECR	O Amazon Elastic Container Registry (Amazon ECR) é um serviço AWS gerenciado de registro Docker seguro, escalável e confiável.	21 de dezembro de 2015

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.