Amazon ECR

Guia do usuário

Versão da API 2015-09-21

Amazon ECR: Guia do usuário

Table of Contents

O que é Amazon Elastic Container Registry?

Amazon Elastic Container Registry (Amazon ECR) é uma ferramenta AWS o serviço de registo de imagens de contentores que é seguro, escalável e fiável. Amazon ECR suporta repositórios de imagens de recipientes privados com permissões baseadas em recursos utilizando AWS IAM para que utilizadores específicos ou Amazon EC2 podem aceder a repositórios e imagens. Os programadores podem usar a sua CLI preferida para empurrar, puxar e gerir imagens Docker, imagens Open Container Initiative (OCI) e artefactos compatíveis com OCI.

A equipe de serviços de contêiner da AWS mantém um roteiro público no GitHub. Ele contém informações sobre no que as equipes estão trabalhando e permite que todos os clientes da AWS forneçam feedback diretamente. Para obter mais informações, consulte Roteiro de contêineres da AWS.

Componentes do Amazon ECR

O Amazon ECR contém os seguintes componentes:

Registro

Um registro do Amazon ECR é fornecido a cada conta da AWS. Você pode criar repositórios de imagens em seu registro e armazenar imagens neles. Para obter mais informações, consulte Amazon ECR registos (p. 14).

Token de autorização

O seu cliente tem de se autenticar para Amazon ECR registos como um AWS antes de poder empurrar e puxar imagens. Para obter mais informações, consulte Autenticação do registo (p. 14).

Repositório

máquinas Amazon ECR O repositório de imagens contém as suas imagens Docker, imagens Open Container Initiative (OCI) e artefactos compatíveis com OCI. Para obter mais informações, consulte Amazon ECR repositórios (p. 17).

Política de repositório

Você pode controlar o acesso aos repositórios e às imagens contidas neles com as políticas de repositório. Para obter mais informações, consulte Políticas de repositório (p. 20).

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 Usar imagens do Amazon ECR com o Amazon ECS (p. 55) e Usar imagens do Amazon ECR com o Amazon EKS (p. 56).

Recursos do Amazon ECR

O Amazon ECR fornece os seguintes recursos:

- As políticas do ciclo de vida permitem-lhe gerir o ciclo de vida das imagens nos seus repositórios. Defini regras que resultam na limpeza de imagens não utilizadas que pode testar antes de aplicar ao seu repositório. Para obter mais informações, consulte Políticas de ciclo de vida (p. 37).
- A digitalização de imagens ajuda a identificar vulnerabilidades de software nas suas imagens de recipiente. Cada repositório pode ser configurado para leitura imediatao que garante que cada nova imagem enviada para o repositório é digitalizada. Pode então recuperar os resultados da digitalização da imagem. Para obter mais informações, consulte Verificação de imagens (p. 50).

Como começar a usar o Amazon ECR

Para usar o Amazon ECR, você precisa estar configurado para instalar a AWS Command Line Interface e o Docker. Para obter mais informações, consulte Configuração com o Amazon ECR (p. 3) e Conceitos básicos do Amazon ECR usando a AWS CLI (p. 8).

Definição de preços do Amazon ECR

com Amazon ECR, paga apenas pela quantidade de dados que armazena nos seus repositórios e pela transferência de dados a partir das suas imagens pushs and pulls. Para obter mais informações, consulte Definição de preços do Amazon ECR.

Configuração com o Amazon ECR

Caso já esteja cadastrado na AWS e usando o Amazon Elastic Container Service (Amazon ECS) ou o Amazon Elastic Kubernetes Service (Amazon EKS), você está próximo de poder usar o Amazon ECR. O processo de configuração desses dois serviços é semelhante, já que o Amazon ECR é uma extensão deles. Para usar a AWS CLI com o Amazon ECR, você deve usar uma versão da AWS CLI que ofereça suporte aos recursos mais recentes do Amazon ECR. Se você sentir falta do suporte a um determinado recurso do Amazon ECR na AWS CLI, atualize para a versão mais recente. Para obter mais informações, consulte http://aws.amazon.com/cli/.

Conclua as seguintes tarefas para se preparar para enviar uma imagem de contêiner ao Amazon ECR pela primeira vez. Caso já tenha concluído qualquer uma dessas etapas, você poderá ignorá-las e passar para a próxima.

Cadastre-se no AWS

Quando você se cadastra na AWS, sua conta da AWS é automaticamente cadastrada em todos os serviços, inclusive o Amazon ECR. Você será cobrado apenas pelos serviços que usar.

Se já tiver uma conta da AWS, passe para a próxima tarefa. Se você ainda não possuir uma conta da AWS, use o procedimento a seguir para criar uma.

Para criar uma conta da AWS

1. Abra https://portal.aws.amazon.com/billing/signup.
2. Siga as instruções online.

   Parte do procedimento de cadastro envolve uma chamada telefônica e a digitação de um código de verificação usando o teclado do telefone.

Anote o número da conta da AWS, pois você precisará dele na próxima tarefa.

Criar um usuário do IAM

Os serviços da AWS, como o Amazon ECR, exigem que você forneça credenciais ao acessá-los, para que o serviço possa determinar se você tem permissão para acessar seus recursos. O console requer sua senha. Você pode criar chaves de acesso para sua conta da AWS para acessar a interface de linha de comando ou a API. Contudo, não recomendamos que você acesse a usando as credenciais da sua conta da AWS; em vez disso, recomendamos que você use o AWS (AWS Identity and Access Management). Crie um usuário do IAM e, em seguida, adicione o usuário a um grupo do IAM com permissões administrativas ou conceda permissões administrativas a esse usuário. Em seguida, você poderá acessar o usando um URL especial e as credenciais do usuário do AWS.

Se você tiver se cadastrado na AWS, mas não criou um usuário do IAM para você mesmo, poderá criar um usando o console do IAM.

Para criar um usuário administrador para você mesmo e adicionar o usuário a um grupo de administradores (console)

1. Entre no console do IAM como o proprietário da conta escolhendo usuário raiz e inserindo o endereço de e-mail de sua da conta AWS. Na próxima página, insira sua senha.

> **Note**
>
> Recomendamos que você siga as melhores práticas para utilizar o usuário do **`Administrator`** IAM abaixo e armazene as credenciais do usuário raiz com segurança. Cadastre-se como usuário raiz para executar somente algumas tarefas de gerenciamento de serviços e contas.

2. No painel de navegação, escolha Usuários e depois Adicionar usuário.
3. Em User name (Nome do usuário), digite **`Administrator`**.
4. Marque a caixa de seleção ao lado de Console de gerenciamento da AWS access (Acesso ao &console;). Então, selecione Custom password (Senha personalizada), e insira sua nova senha na caixa de texto.
5. (Opcional) Por padrão, a AWS exige que o novo usuário crie uma senha ao fazer login pela primeira vez. Você pode desmarcar a caixa de seleção próxima de User must create a new password at next sign-in (O usuário deve criar uma senha no próximo login) para permitir que o novo usuário redefina a senha depois de fazer login.
6. Escolha Próximo: Permissões.
7. Em Set permissions (Conceder permissões), escolha Add user to group (Adicionar usuário ao grupo).
8. Escolha Create group (Criar grupo).
9. Na caixa de diálogo Create group (Criar grupo), em Group name (Nome do grupo), digite **`Administrators`**.
10. Escolha Filter policies (Filtrar políticas) e, depois, selecione AWS managed -job function (Função de trabalho gerenciado pela &AWS;) para filtrar o conteúdo de tabelas.
11. Na lista de políticas, marque a caixa de seleção AdministratorAccess. A seguir escolha Criar grupo.

    > **Note**
    >
    > Você deve ativar o acesso de usuário e função do IAM ao faturamento para poder usar as permissões do `AdministratorAccess` a fim de acessar o console do AWS Billing and Cost Management. Para fazer isso, siga as instruções na etapa 1 do tutorial sobre como delegar acesso ao console de faturamento.

12. Suporte a lista de grupos, selecione a caixa de seleção para seu novo grupo. Escolha Refresh (Atualizar) caso necessário, para ver o grupo na lista.
13. Escolha Next: Tags (Próximo: tags).
14. (Opcional) Adicione metadados ao usuário anexando tags como pares de chave-valor. Para obter mais informações sobre como usar tags no IAM, consulte Marcar entidades do IAM no Guia do usuário do IAM.
15. Escolha Next: Review (Próximo: Análise) para ver uma lista de associações de grupos a serem adicionadas ao novo usuário. Quando você estiver pronto para continuar, selecione Criar usuário.

Você pode usar esse mesmo processo para criar mais grupos e usuários e conceder aos seus usuários acesso aos recursos de sua conta da AWS. Para saber como usar políticas para restringir as permissões de usuário a recursos específicos da AWS, acesse Gerenciamento de acesso e Políticas de exemplo.

Para fazer login como esse novo usuário do IAM, faça logout no console da AWS e use a seguinte URL, em que your_aws_account_id é o número de sua conta da AWS sem hifens (por exemplo, se o número da conta da AWS é `1234-5678-9012`, o ID da conta da AWS é `123456789012`):

```
https://your_aws_account_id.signin.aws.amazon.com/console/
```

Insira o nome e a senha de usuário do IAM que você acabou de criar. Quando você está conectado, a barra de navegação exibe "your_user_name @ your_aws_account_id".

Se você não quiser que o URL da página de cadastro contenha o ID da sua conta da AWS, crie um alias da conta. No painel do IAM, escolha Customize (Personalizar) e insira um Account Alias (Alias da conta), por exemplo, o nome da sua empresa. Para obter mais informações, consulte O ID da sua conta da AWS e seu alias no Guia do usuário do IAM.

Para fazer o login depois de criar o alias de uma conta, use o seguinte URL:

```
https://your_account_alias.signin.aws.amazon.com/console/
```

Para verificar o link de login para usuários do IAM em sua conta, abra o console do IAM e procure em IAM users sign-in link (Link de login de usuários do IAM) no painel.

Para obter mais informações sobre o IAM, consulte o Guia do usuário do AWS Identity and Access Management.

Conceitos básicos do Amazon ECR usando o Console de gerenciamento da AWS

Comece a usar o Amazon ECR criando um repositório no console do Amazon ECR. O console do Amazon ECR orienta você no processo para começar a criar seu primeiro repositório.

Antes de começar, você deve concluir as etapas em Configuração com o Amazon ECR (p. 3).

Como criar um repositório de imagens

Um repositório é o local em que armazena as imagens do Docker ou da Open Container Initiative (OCI) no Amazon ECR. Sempre que enviar ou receber uma imagem do Amazon ECR, especifique o repositório e o local do registro, que informa para onde enviar a imagem ou de onde retirá-la.

1. Abra o console do Amazon ECR em https://console.aws.amazon.com/ecr/.
2. Escolha Get Started.
3. 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 substituídas. Para obter mais informações, consulte Mutabilidade de tag de imagem (p. 49).
4. Em Scan on push (Verificar ao enviar), escolha a configuração de verificação de imagens para o repositório. Os repositórios configurados para verificar ao enviar iniciarão uma verificação de imagem sempre que uma imagem for enviada, caso contrário, as verificações de imagem deverão ser iniciadas manualmente. Para obter mais informações, consulte Verificação de imagens (p. 50).
5. Escolha Create repository (Criar repositório).

Criar, marcar e enviar uma imagem do Docker por push

Nesta seção do assistente, você usa a CLI do Docker para marcar uma imagem local existente (que você tiver criado a partir de um Dockerfile ou enviado por push de outro registro, como o Docker Hub) e, em seguida, enviar a imagem marcada para seu registro do Amazon ECR. Para obter etapas mais detalhadas sobre como usar a CLI do Docker, consulte Conceitos básicos do Amazon ECR usando a AWS CLI (p. 8).

1. Selecione o repositório criado e escolha View push commands (Exibir comandos de push) para ver as etapas de como enviar uma imagem para o novo repositório.
2. Recupere o comando docker login que você possa usar para autenticar o cliente Docker no registro colando o comando aws ecr get-login do console em uma janela de terminal.

   Note

   O comando get-login está disponível na AWS CLI desde a versão 1.9.15, mas recomendamos a versão 1.11.91 ou posterior para versões recentes do Docker (17.06 ou posterior). Você pode verificar a versão da AWS CLI com o comando aws --version. Se você estiver usando a versão 17.06 do Docker ou posterior, inclua a opção `--no-include-email` após `get-login`. Se você receber um erro `Unknown options: --no-include-email`, instale a versão mais recente da CLI da AWS. Para obter mais informações, consulte Instalar a interface de linha de comando da AWS no Guia do usuário do AWS Command Line Interface.

3. Execute o comando docker login retornado na etapa anterior. Esse comando fornece um token de autorização válido por 12 horas.

   Important

   Durante a execução desse comando docker login, a string de comando pode ser visível a outros usuários no sistema em uma exibição da lista de processos (ps -e). Como o comando docker login contém credenciais de autenticação, há risco de que outros usuários no sistema possam visualizá-las. Eles podem usar as credenciais para conseguir acesso de envio aos repositórios. Se você não estiver em um sistema seguro, considere esses riscos e efetue login interativamente ao omitir a opção `-p` *`password`* e forneça a senha quando solicitado.
4. (Opcional) Se você tiver um Dockerfile para a imagem a ser enviada, compile a imagem e marque-a para o novo repositório. Cole o comando docker build do console em uma janela do terminal. Certifique-se de que você esteja no mesmo diretório do Dockerfile.
5. Marque a imagem do registro ECR e do novo repositório colando o comando docker tag do console em uma janela de terminal. O comando do console pressupõe que a imagem tenha sido compilada com base em um Dockerfile na etapa anterior. Se não você tiver compilado a imagem com base em um Dockerfile, substitua a primeira instância de *`repository`*`:latest` pelo ID da imagem ou o nome da imagem local a ser enviada.
6. Envie a imagem recém-marcada para o repositório ECR colando o comando docker push em uma janela de terminal.
7. Escolha Close (Fechar).

Conceitos básicos do Amazon ECR usando a AWS CLI

As etapas a seguir orientam você pelas etapas necessárias para enviar uma imagem de contêiner ao Amazon ECR pela primeira vez usando a CLI do Docker e a AWS CLI.

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

Pré-requisitos

Antes de começar, você deve concluir as etapas em Configuração com o Amazon ECR (p. 3).

Se você ainda não tiver o Docker e a AWS CLI mais recentes instalados e preparados, use as etapas a seguir para instalar essas duas ferramentas.

Instalar a AWS CLI

Você pode usar as ferramentas de linha de comando da AWS para emitir comandos na linha de comando do seu sistema e realizar tarefas do Amazon ECR e da AWS. Isso pode ser mais rápido e mais conveniente do que usar o console. As ferramentas da linha de comando também são úteis para criar scripts que executam tarefas da AWS.

Para usar a AWS CLI com o Amazon ECR, instale a versão mais recente da AWS CLI (a funcionalidade do Amazon ECR está disponível na AWS CLI a partir da versão 1.9.15). Você pode verificar a versão da AWS CLI com o comando aws --version. Para obter informações sobre como instalar a AWS CLI ou atualizá-la para a versão mais recente, consulte Instalar a AWS CLI versão 2 no Guia do usuário do 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 Ubuntu, e até o Mac OSX e o 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 executar uma instância do Amazon Linux 2 e instalar o Docker para começar.

Se você já tiver um Docker instalado, vá para Etapa 1: criar uma imagem do Docker (p. 9).

Para instalar o Docker em uma instância do Amazon EC2

1. Execute uma instância com a AMI do Amazon Linux 2. Para obter mais informações, consulte Executar de uma instância no Guia do usuário do Amazon EC2 para instâncias do Linux.
2. Conecte-se à sua instância. Para obter mais informações, consulte Conectar-se à instância do Linux no Guia do usuário do Amazon EC2 para instâncias do Linux.

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 amazon-linux-extras 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`. Você pode 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

As definições de tarefa do Nesta seção, crie uma imagem de docker de um aplicativo web simples e teste-a no seu sistema local ou instância do EC2. Em seguida, envie a imagem ao registro de contêiner (como o Amazon ECR ou o Docker Hub) para poder usá-la em uma definição de tarefa do ECS.

Para criar uma imagem do Docker de um aplicativo web simples

1. Crie um arquivo chamado `Dockerfile`. 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 ubuntu:18.04

# Install dependencies
RUN apt-get update && \
 apt-get -y install apache2
```

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

# Configure apache
RUN echo '. /etc/apache2/envvars' > /root/run_apache.sh && \
 echo 'mkdir -p /var/run/apache2' >> /root/run_apache.sh && \
 echo 'mkdir -p /var/lock/apache2' >> /root/run_apache.sh && \
 echo '/usr/sbin/apache2 -D FOREGROUND' >> /root/run_apache.sh && \
 chmod 755 /root/run_apache.sh

EXPOSE 80

CMD /root/run_apache.sh
```

Esse Dockerfile usa a imagem do Ubuntu 18.04. As instruções de `RUN` atualizam os caches do pacote, instalam alguns pacotes de software para o servidor web e, em seguida, gravam o conteúdo "Hello World!" na raiz do documento do servidor web. A instrução `EXPOSE` expõe a porta 80 no 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. Execute docker images para verificar se a imagem foi criada corretamente.

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

   Resultado:

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

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. Você pode ignorar a mensagem "`Could not reliably determine the server's 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, localize o endereço IP da VM VirtualBox que está hospedando o Docker com o comando docker-machine ip, substituindo *machine-name* pelo nome da máquina de docker que você está usando.

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

Você deverá ver uma página da web com a sua instrução "Hello World!".

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 o CLI do Docker para seu registro padrão. Desta forma, o comando docker pode adicionar e extrair imagens com Amazon ECR. A AWS CLI fornece um comando get-login-password 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 no 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 interface de linha de comando da AWS no Guia do usuário do AWS Command Line Interface.

- 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
```

- Get-ECRLoginCommand (AWS Tools para 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 você tem uma imagem para enviar ao Amazon ECR, precisa criar um repositório para guardá-la. Neste exemplo, você cria um repositório chamado `hello-world` 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-world \
    --image-scanning-configuration scanOnPush=true \
    --region us-east-1
```

Etapa 4: enviar uma imagem ao Amazon ECR

Agora você pode enviar a imagem ao repositório do Amazon ECR que criou na seção anterior. Você pode usar a CLI do docker para enviar imagens, mas há alguns pré-requisitos que devem ser atendidos para que isso funcione corretamente:

- A versão mínima do docker está instalada: 1.7

- O token de autorização do Amazon ECR foi configurado com docker 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
```

   Resultado:

```
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.us-east-1.amazonaws.com/hello-
world:latest
```

3. Envie a imagem.

```
docker push aws_account_id.dkr.ecr.us-east-1.amazonaws.com/hello-world:latest
```

   Resultado:

```
The push refers to a repository [aws_account_id.dkr.ecr.us-east-1.amazonaws.com/hello-
world] (len: 1)
e9ae3c220b23: Pushed
a6785352b25c: Pushed
0998bf8fb9e9: Pushed
0a85502c06c9: Pushed
latest: digest: sha256:215d7e4121b30157d8839e81c4e0912606fca105775bb0636b95aed25f52c89b
 size: 6774
```

Etapa 5: extrair uma imagem do Amazon ECR

Depois que a imagem for enviada ao repositório do Amazon ECR, você poderá extraí-la de outros locais. Use a CLI docker para extrair imagens, mas há alguns pré-requisitos que devem ser atendidos para que isso funcione corretamente:

- A versão mínima do docker está instalada: 1.7
- O token de autorização do Amazon ECR foi configurado com docker 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.us-east-1.amazonaws.com/hello-world:latest
```

Resultado:

```
latest: Pulling from hello-world
0a85502c06c9: Pull complete
0998bf8fb9e9: Pull complete
a6785352b25c: Pull complete
e9ae3c220b23: Pull complete
Digest: sha256:215d7e4121b30157d8839e81c4e0912606fca105775bb0636b95aed25f52c89b
Status: Downloaded newer image for aws_account_id.dkr.ecr.us-east-1.amazonaws.com/hello-
world:latest
```

Etapa 6: excluir uma imagem

Se você decidir que não precisa ou não quer mais uma imagem em um dos repositórios, poderá excluí-la com o comando batch-delete-image. Para excluir uma imagem, você deve especificar o repositório em que ele está e um valor `imageTag` ou `imageDigest` para imagem. O exemplo abaixo exclui uma imagem no repositório `hello-world` com a tag de imagem `latest`.

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

Resultado:

```
{
    "failures": [],
    "imageIds": [
        {
            "imageTag": "latest",
            "imageDigest":
 "sha256:215d7e4121b30157d8839e81c4e0912606fca105775bb0636b95aed25f52c89b"
        }
    ]
}
```

Etapa 7: excluir um repositório

Se você decidir que não precisa ou não quer mais um repositório inteiro de imagens, exclua o repositório. Por padrão, não é possível excluir um repositório que contém imagens. No entanto, o sinalizador `--force` permite isso. Para excluir um repositório que contém imagens (e todas as imagens contidas nele), execute o seguinte comando.

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

Amazon ECR registos

Amazon ECROs registros do hospedam as imagens do contêiner em uma arquitetura altamente disponível e escalável, o que permite que você implante contêineres para seus aplicativos de modo confiável. É possível usar o registro para gerenciar repositórios de imagens que consistem em imagens do Docker e da Open Container Initiative (OCI). Cada AWS a conta é fornecida com um único (predefinição) Amazon ECR registo.

Conceitos de registo

- O URL do seu registro padrão é `https://aws_account_id.dkr.ecr.region.amazonaws.com`.
- Por padrão, sua conta tem acesso de leitura e gravação aos repositórios no seu registro padrão (). No entanto, IAM os utilizadores necessitam de autorização para efectuar chamadas para Amazon ECR API e para empurrar ou puxar imagens dos seus repositórios. Amazon ECR fornece várias políticas geridas para controlar o acesso do utilizador a níveis variáveis. Para obter mais informações, consulte Amazon Elastic Container RegistryExemplos de políticas baseadas em identidade do (p. 69).
- Tem de autenticar o cliente do acoplador para um registo para que possa utilizar o docker push e docker pull comandos para empurrar e puxar imagens de e para os repositórios nesse registo. Para obter mais informações, consulte Autenticação do registo (p. 14).
- Os repositórios podem ser controlados com políticas de acesso de usuários do IAM e políticas de repositório. Para obter mais informações sobre políticas de repositórios, consulte Políticas de repositório (p. 20).

Autenticação do registo

Pode utilizar o Console de gerenciamento da AWS, o AWS CLI, ou o AWS para criar e gerir repositórios. Você também pode usar esses métodos para realizar algumas ações em imagens, como listá-las ou excluí-las. Estes clientes utilizam padrão AWS métodos de autenticação. Embora tecnicamente 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 oferece suporte a métodos de autenticação do IAM nativos. é necessário seguir etapas adicionais para que o Amazon ECR possa autenticar e autorizar solicitações push e pull do Docker.

Os métodos de autenticação de registro a seguir estão disponíveis.

Utilizar o Amazon ECR auxiliar de credencial

Amazon ECRO 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 etapas de instalação e configuração, consulte Auxiliar de credencial do acoplador ECR da Amazon.

Utilizar um token de autorização

O escopo de permissão de um token de autorização corresponde ao principal do IAM usado para recuperar o token de autenticação. É utilizado um token de autenticação para aceder a qualquer

Amazon ECR que o seu IAM o principal tem acesso e é válido durante 12 horas. Para obter um token de autorização, tem de utilizar o idtauthorizationtoken Operação API para recuperar um token de autorização de base64 com o nome de utilizador `AWS` e uma palavra-passe codificada. O AWS CLI `get-login-password` o comando simplifica-o, recuperando e descodificando o token de autorização, o qual pode depois fazer o tubo para um docker login para autenticar.

Para autenticar o acoplador a um Amazon ECR registo com a palavra-passe 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 no 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 interface de linha de comando da AWS no Guia do usuário do AWS Command Line Interface.

- 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
```

- Get-ECRLoginCommand (AWS Tools para Windows PowerShell)

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

Como autenticar o Docker em um registro do Amazon ECR com get-login

Ao usar versões da AWS CLI anteriores à 1.17.10, o comando get-login fica disponível para autenticação no registro do Amazon ECR. Você pode verificar a versão da AWS CLI com o comando aws --version.

1. Execute o comando aws ecr get-login. O exemplo abaixo indica como fazer o registro padrão associado à conta que fez a solicitação. Para acessar os registros de outras contas, use a opção `--registry-ids` *aws_account_id*. Para obter mais informações, consulte get-login no AWS CLI Command Reference.

   ```
   aws ecr get-login --region region --no-include-email
   ```

   A saída resultante é um comando docker login usado para autenticar o cliente do Docker no registro do Amazon ECR.

   ```
   docker login -u AWS -p password https://aws_account_id.dkr.ecr.region.amazonaws.com
   ```

2. Copie e cole o comando docker login em um terminal para autenticar a CLI do Docker no registro. Este comando fornece um token de autorização válido por 12 horas para o registro especificado.

   > **Note**
   >
   > Se você estiver usando o Windows PowerShell, não é possível copiar e colar strings longas como essa. Use o seguinte comando.

   ```
   Invoke-Expression -Command (Get-ECRLoginCommand -Region region).Command
   ```

Important

Durante a execução desse comando docker login, a string de comando pode ser visível a outros usuários no sistema em uma exibição da lista de processos (ps -e). Como o comando docker login contém credenciais de autenticação, há risco de que outros usuários no sistema possam visualizá-las. Eles podem usar as credenciais para conseguir acesso de envio aos repositórios. Se você não estiver em um sistema seguro, considere esses riscos e efetue login interativamente ao omitir a opção `-p` *`password`* e forneça a senha quando solicitado.

Utilizar a autenticação API HTTP

Amazon ECR apoia o API HTTP do registo do acoplador. No entanto, como o Amazon ECR é um registro privado, você deve fornecer um token de autorização com cada solicitação HTTP. Pode adicionar um cabeçalho de autorização HTTP utilizando o `-H` opção para curl e passe o token de autorização fornecido pelo get-authorization-token AWS CLI comando.

Para autenticar com o Amazon ECR API HTTP

1. Recupere um token de autorização com a AWS CLI e configure-o para uma variável de ambiente.

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

2. Para autenticar para a API, passe o `$TOKEN` variável para o `-H` opção de curl. Por exemplo, o seguinte comando lista as etiquetas de imagem num Amazon ECR repositório. Para mais informações, consulte o API HTTP do registo do acoplador documentação de referência.

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

Resultado

```
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"]}
```

Amazon ECR repositórios

O Amazon Elastic Container Registry (Amazon ECR) fornece operações de API para criar, monitorar e excluir imagem e definir permissões de repositório que controlam quem pode acessá-los. É possível executar as mesmas ações na seção Repositories (Repositórios) do console do Amazon ECR. O Amazon ECR também se integra à CLI do Docker, permitindo que você envie e extraia imagens de seus ambientes de desenvolvimento para os repositórios.

Tópicos

Conceitos de repositório

- 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 do IAM necessitam de permissões para fazer chamadas à APIs do Amazon ECR e para enviar e extrair imagens de seus repositórios. O Amazon ECR fornece várias políticas gerenciadas para controlar o acesso do usuário em diversos níveis. Para obter mais informações, consulte Amazon Elastic Container RegistryExemplos de políticas baseadas em identidade do (p. 69).
- Os repositórios podem ser controlados com políticas de acesso de usuários do IAM e políticas de repositório. Para obter mais informações, consulte Políticas de repositório (p. 20).
- 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`. Cada equipe poderia ter sua própria imagem chamada `web-app`, mas como cada uma delas é precedida pelo namespace da equipe, as duas imagens podem ser usadas simultaneamente sem interferência. A imagem da Equipe A se chamaria `team-a/web-app`, e a imagem da Equipe B se chamaria `team-b/web-app`.

Criar um repositório

Antes de enviar suas imagens de Docker ao Amazon ECR, você precisa criar um repositório de armazená-las. É possível criar repositórios do Amazon ECR usando o Console de gerenciamento da AWS, a AWS CLI ou os SDKs da AWS.

Como criar um repositório do

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), selecione Create repository (Criar repositório).

5. Em Repository name (Nome do repositório), insira um nome exclusivo para o repositório. O nome do repositório pode ser especificado sozinho (como `nginx-web-app`) ou pode ser prefixado com um namespace para agrupar o repositório em uma categoria (como `project-a/nginx-web-app`).

   Note

   O nome tem de começar por uma letra e só pode conter letras minúsculas, números, hífenes, sublinhados e barras.
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 substituídas. Para obter mais informações, consulte Mutabilidade de tag de imagem (p. 49).
7. Em Scan on push (Verificar ao enviar), escolha a configuração de verificação de imagens para o repositório. Os repositórios configurados para verificar ao enviar iniciarão uma verificação de imagem sempre que uma imagem for enviada, caso contrário, as verificações de imagem deverão ser iniciadas manualmente. Para obter mais informações, consulte Verificação de imagens (p. 50).
8. para Encriptação KMS, escolha se pretende ativar a encriptação das imagens no repositório utilizando AWS Key Management Service. Por predefinição, quando a encriptação KMS está ativada Amazon ECR usa um AWS chave mestre de cliente gerida (CMK) com nome alternativo `aws/ecr`, que é criado na sua conta na primeira vez que cria um repositório com encriptação KMS ativada. Para obter mais informações, consulte Criptografia em repouso (p. 75).
9. Quando a encriptação KMS estiver ativada, selecione Definições de encriptação de clientes (avançadas) para escolher o seu próprio CMK. O CMK tem de existir na mesma região do cluster. Escolher Criar uma AWS KMS chave para navegar até à AWS KMS para criar a sua própria chave.
10. Escolha Create repository (Criar repositório).
11. (Opcional) Selecione o repositório criado e escolha View push commands (Visualizar comandos de push) para ver as etapas a seguir para enviar uma imagem ao novo repositório.

    a. Recupere o comando docker login que você possa usar para autenticar o cliente Docker no registro colando o comando aws ecr get-login do console em uma janela de terminal.

       Note

       O comando get-login está disponível na AWS CLI desde a versão 1.9.15, mas recomendamos a versão 1.11.91 ou posterior para versões recentes do Docker (17.06 ou posterior). Você pode verificar a versão da AWS CLI com o comando aws --version. Se você estiver usando a versão 17.06 do Docker ou posterior, inclua a opção `--no-include-email` após `get-login`. Se você receber um erro `Unknown options: --no-include-email`, instale a versão mais recente da CLI da AWS. Para obter mais informações, consulte Instalar a interface de linha de comando da AWS no Guia do usuário do AWS Command Line Interface.

    b. Execute o comando docker login retornado na etapa anterior. Esse comando fornece um token de autorização válido por 12 horas.

       Important

       Durante a execução desse comando docker login, a string de comando pode ser visível a outros usuários no sistema em uma exibição da lista de processos (ps -e). Como o comando docker login contém credenciais de autenticação, há risco de que outros usuários no sistema possam visualizá-las. Eles podem usar as credenciais para conseguir acesso de envio aos repositórios. Se você não estiver em um sistema seguro, considere esses riscos e efetue login interativamente ao omitir a opção `-p` *`password`* e forneça a senha quando solicitado.

    c. (Opcional) Se você tiver um Dockerfile para a imagem a ser enviada, compile a imagem e marque-a para o novo repositório. Cole o comando docker build do console em uma janela do terminal. Certifique-se de que você esteja no mesmo diretório do Dockerfile.

    d. Marque a imagem do registro ECR e do novo repositório colando o comando docker tag do console em uma janela de terminal. O comando do console pressupõe que a imagem tenha sido

compilada com base em um Dockerfile na etapa anterior. Se não você tiver compilado a imagem com base em um Dockerfile, substitua a primeira instância de *repository*:latest pelo ID da imagem ou o nome da imagem local a ser enviada.

e. Envie a imagem recém-marcada para o repositório ECR colando o comando docker push em uma janela de terminal.

f. Escolha Close (Fechar).

Visualizar informações do repositório

Depois de criar um repositório, você poderá ver as informações dele no Console de gerenciamento da AWS:

- Quais imagens são armazenadas em um repositório
- Se uma imagem é marcada
- As tags da imagem
- O resumo SHA das imagens
- O tamanho das imagens em MiB
- Quando a imagem foi extraída 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 e, portanto, ele pode retornar um tamanho de imagem maior do que aqueles mostrados no Console de gerenciamento da AWS.

Para visualizar as informações do 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 a ser visualizado.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositórios, escolha o repositório a ser visualizado.
5. Em Repositórios: ***repository_name*** , utilize a barra de navegação para ver informações sobre uma imagem.
   - Escolha Imagens para visualizar informações sobre as imagens no repositório. Se houver imagens não marcadas que você deseja excluir, selecione a caixa à esquerda dos repositórios a serem excluídos e escolha Excluir. Para obter mais informações, consulte Excluir uma imagem (p. 35).
   - Escolha Permissões para visualizar as políticas de repositório aplicadas ao repositório. Para obter mais informações, consulte Políticas de repositório (p. 20).
   - 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 obter mais informações, consulte Políticas de ciclo de vida (p. 37).
   - Selecione Tags para visualizar as tags de metadados que são aplicadas ao repositório.

Editar um repositório

Os repositórios existentes podem ser editados para alterar a mutabilidade da tag de imagem e as configurações de verificação de imagem.

Para editar um 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 a ser editado.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositories (Repositórios), selecione o repositório a ser excluído e escolha Edit (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 substituídas. Para obter mais informações, consulte Mutabilidade de tag de imagem (p. 49).
6. Em Scan on push (Verificar ao enviar), escolha a configuração de verificação de imagens para o repositório. Os repositórios configurados para verificar ao enviar iniciarão uma verificação de imagem sempre que uma imagem for enviada, caso contrário, as verificações de imagem deverão ser iniciadas manualmente. Para obter mais informações, consulte Verificação de imagens (p. 50).
7. Escolha Save (Salvar) para atualizar as configurações do repositório.

Excluir um repositório

Se você já terminou de usar um repositório, pode excluí-lo. Quando você exclui um repositório no Console de gerenciamento da AWS, todas as imagens contidas nele também são excluídas. Essa ação não pode ser desfeita.

Para excluir um 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 a ser excluído.
3. No painel de navegação, escolha Repositories (Repositórios).
4. Na página Repositórios, selecione o repositório para excluir e escolha Excluir.
5. Na janela Eliminar ***repository_name*** , verifique se os repositórios seleccionados devem ser eliminados e escolha Eliminar.

   Important

   Quaisquer imagens nos repositórios selecionados também serão excluídas.

Políticas de repositório

Amazon ECRO usa permissões baseadas em recursos para controlar o acesso a repositórios. As permissões baseadas em recursos permitem especificar quais funções ou usuários do IAM têm acesso a um repositório e quais ações podem realizar nele. Por padrão, somente o proprietário do repositório tem acesso a ele. Você pode aplicar um documento de política que concede permissões adicionais ao repositório.

Políticas de repositório vs. IAM políticas

Amazon ECR as políticas do repositório são um subconjunto de IAM políticas que são trocadas e utilizadas especificamente para controlar o acesso a Amazon ECR repositórios. IAM as políticas são geralmente utilizadas para aplicar permissões para toda a Amazon ECR mas também pode ser utilizado para controlar o acesso a recursos específicos.

Ambos Amazon ECR políticas de repositório e IAM as políticas são utilizadas ao determinar quais as acções que uma IAM o utilizador ou a função pode realizar num repositório. Se uma função ou um usuário tiver permissão para executar uma ação por meio de uma política de repositório, mas tem 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 concedam permissões para a API `ecr:GetAuthorizationToken` por meio de uma política do IAM antes que eles possam fazer a autenticação em um registro e enviar ou extrair qualquer imagem de um repositório do Amazon ECR. O Amazon ECR fornece várias políticas gerenciadas do IAM para controlar o acesso de usuários em níveis variados. Para obter mais informações, consulte Amazon Elastic Container RegistryExemplos de políticas baseadas em identidade do (p. 69).

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 um Amazon ECR política do repositório, que permite uma IAM utilizador para descrever o repositório e as imagens dentro do repositório.

```
{
  "Version": "2008-10-17",
  "Statement": [{
    "Sid": "ECR Repository Policy",
    "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 Resources (p. 65).

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "ECR Repository Policy",
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::account-id:user/username"
    },
    "Action": [
      "ecr:DescribeImages",
      "ecr:DescribeRepositories"
    ],
    "Resource": [
      "arn:aws:ecr:region:account-id:repository/repository-name"
    ]
    }]
}
```

Tópicos

- Definir uma declaração de política de repositório (p. 22)
- Eliminar uma declaração de política de repositório (p. 22)
- Exemplos de política de repositório (p. 23)

Definir uma declaração de política de repositório

Você pode adicionar uma instrução da política de acesso a um repositório no Console de gerenciamento da AWS seguindo as etapas abaixo. Você pode adicionar várias instruções de política por repositório. Para obter exemplos de políticas do , consulte Exemplos de política de repositório (p. 23).

> **Important**
>
> O Amazon ECR exige que os usuários concedam permissões para a API `ecr:GetAuthorizationToken` por meio de uma política do IAM antes que eles possam fazer a autenticação em um registro e enviar ou extrair qualquer imagem de um repositório do Amazon ECR. O Amazon ECR fornece várias políticas gerenciadas do IAM para controlar o acesso de usuários em níveis variados. Para obter mais informações, consulte Amazon Elastic Container RegistryExemplos de políticas baseadas em identidade do (p. 69).

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 Repositórios.
4. No Repositórios , escolha o repositório para definir uma declaração de política em.
5. No painel de navegação, escolha Permissões, Editar.
6. No Editar permissões página, escolher Adicionar declaração.
7. Para Nome da declaração, introduza um nome para a declaração.
8. Para Efeito, escolha se a declaração da apólice resultará numa autorização ou recusa explícita.
9. Para Principal, escolha o âmbito para aplicar a declaração de política a. Para mais informações, consulte Elementos da política JSON AWS: Principal no Guia do usuário do IAM.
   - Pode aplicar a declaração a todos os autenticados AWS seleccionando o Todos (*).
   - Para Responsável de serviço, especifique o nome principal do serviço (por exemplo, `ecs.amazonaws.com`) para aplicar a declaração a um serviço específico.
   - Para ID de conta AWS, especifique um AWS número de conta (por exemplo, `111122223333`) para aplicar a declaração a todos os utilizadores sob um AWS conta. Várias contas podem ser especificadas usando uma lista delimitada por vírgulas.
   - Para IAM Entidades, selecione as funções ou utilizadores sob o seu AWS para aplicar a declaração a.

     > **Note**
     >
     > Para políticas de repositório mais complicadas, que não são compatíveis com o Console de gerenciamento da AWS no momento, você pode aplicar a política com o comando set-repository-policy da AWS CLI.
10. Para Ações, escolha o âmbito do Amazon ECR As operações da API que a declaração da política deve aplicar-se a partir da lista de operações individuais da API.
11. Quando terminar, escolha Guardar para definir a política.
12. Repita a etapa anterior para cada política de repositório a ser adicionada.

Eliminar uma declaração de política de repositório

Se você não quiser mais que uma instrução de política de repositório existente seja aplicada a um repositório, exclua-a.

Para excluir 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 do qual será excluída uma instrução de política.
3. No painel de navegação, escolha Repositórios.
4. No Repositórios , escolha o repositório para eliminar uma declaração de política de.
5. No painel de navegação, escolha Permissões, Editar.
6. No Editar permissões página, escolher Eliminar.

Exemplos de política de repositório

Os exemplos seguintes mostram declarações de políticas que pode utilizar para controlar as permissões que os utilizadores têm de Amazon ECR repositórios.

> **Important**
>
> O Amazon ECR exige que os usuários concedam permissões para a API `ecr:GetAuthorizationToken` por meio de uma política do IAM antes que eles possam fazer a autenticação em um registro e enviar ou extrair qualquer imagem de um repositório do Amazon ECR. O Amazon ECR fornece várias políticas gerenciadas do IAM para controlar o acesso de usuários em níveis variados. Para obter mais informações, consulte Amazon Elastic Container RegistryExemplos de políticas baseadas em identidade do (p. 69).

Exemplo: Permitir um IAM utilizador na sua conta

A política de repositório a seguir permite que usuários do IAM na sua conta insiram e extraiam imagens.

```
{
    "Version": "2008-10-17",
    "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:GetDownloadUrlForLayer",
                "ecr:BatchGetImage",
                "ecr:BatchCheckLayerAvailability",
                "ecr:PutImage",
                "ecr:InitiateLayerUpload",
                "ecr:UploadLayerPart",
                "ecr:CompleteLayerUpload"
            ]
        }
    ]
}
```

Exemplo: Permitir outra conta

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

```
{
```

```
    "Version": "2008-10-17",
    "Statement": [
        {
            "Sid": "AllowCrossAccountPush",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::account-id:root"
            },
            "Action": [
                "ecr:GetDownloadUrlForLayer",
                "ecr:BatchCheckLayerAvailability",
                "ecr:PutImage",
                "ecr:InitiateLayerUpload",
                "ecr:UploadLayerPart",
                "ecr:CompleteLayerUpload"
            ]
        }
    ]
}
```

A seguinte política de repositório permite alguns IAM utilizadores para puxar imagens (*pull-user-1* e *pull-user-2*) ao mesmo tempo que fornece acesso total a outro (*admin-user*).

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

```
{
    "Version": "2008-10-17",
    "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:GetDownloadUrlForLayer",
                "ecr:BatchGetImage"
            ]
        },
        {
            "Sid": "AllowAll",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::account-id:user/admin-user"
            },
            "Action": [
                "ecr:*"
            ]
        }
    ]
}
```

Exemplo: Permitir todos AWS contas para puxar imagens

A política de repositório a seguir permite que todas as contas da AWS extraiam imagens.

```
{
    "Version": "2008-10-17",
    "Statement": [
        {
            "Sid": "AllowPull",
            "Effect": "Allow",
            "Principal": "*",
            "Action": [
                "ecr:GetDownloadUrlForLayer",
                "ecr:BatchGetImage"
            ]
        }
    ]
}
```

Exemplo: Recusar todos

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

```
{
    "Version": "2008-10-17",
    "Statement": [
        {
            "Sid": "DenyPull",
            "Effect": "Deny",
            "Principal": "*",
            "Action": [
                "ecr:GetDownloadUrlForLayer",
                "ecr:BatchGetImage"
            ]
        }
    ]
}
```

Exemplo: Restringir o acesso a endereços IP específicos

O exemplo a seguir concede permissões a qualquer usuário para executar qualquer operação do Amazon ECR quando aplicada a um repositório. No entanto, a solicitação deve se originar no intervalo de endereços IP especificados na condição.

A condição nesta instrução identifica o intervalo `54.240.143.*` de endereços IP do protocolo de internet versão 4 (IPv4), com uma exceção: . `54.240.143.188`.

O `Condition` o bloco utiliza o `IpAddress` e `NotIpAddress` e o `aws:SourceIp` tecla de função, que é uma AWS-chave de condição alargada. Para mais informações sobre estas teclas de condição, consulte AWS Teclas de contexto da condição global. Os valores IPv4 `aws:sourceIp` usam a notação CIDR padrão. Para mais informações, consulte Operadores de condições de endereço IP no Guia do usuário do IAM.

```
{
    "Version": "2012-10-17",
    "Id": "ECRPolicyId1",
    "Statement": [
        {
            "Sid": "IPAllow",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "ecr:*",
            "Condition": {
                "NotIpAddress": {
```

```
                    "aws:SourceIp": "54.240.143.188/32"
                },
                "IpAddress": {
                    "aws:SourceIp": "54.240.143.0/24"
                }
            }
        }
    ]
}
```

Exemplo: Função vinculada ao serviço

A seguinte política de repositório permite AWS CodeBuild acesso ao Amazon ECR As ações da API são necessárias para integração com esse serviço. Para mais informações, consulte Amazon ECR Amostra para CodeBuild no Guia do usuário do AWS CodeBuild.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CodeBuildAccess",
            "Effect": "Allow",
            "Principal": {
                "Service": "codebuild.amazonaws.com"
            },
            "Action": [
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer"
            ]
        }
    ]
}
```

Marcar um Amazon ECR repositório

Para o ajudar a gerir o seu Amazon ECR repositórios, pode opcionalmente atribuir os seus próprios metadados a cada repositório sob a forma de etiquetas. Este tópico descreve tags e mostra a você como criá-los.

Tópicos

Conceitos básicos de tags

Uma etiqueta é uma etiqueta que atribui a um AWS recurso. Cada tag consiste num chave e um valor, ambos definem.

As tags permitem categorizar seus recursos do AWS de diferentes formas (como por finalidade, por proprietário ou por ambiente). Isso é útil quando você tem muitos recursos do mesmo tipo — é possível

identificar rapidamente um recurso específico baseado nas tags que você atribuiu a ele. Por exemplo, você pode definir um conjunto de tags para os repositórios do Amazon ECR da sua conta que ajudem a rastrear o proprietário de cada repositório.

Recomendamos que você desenvolva um conjunto de chave de tags que atenda suas necessidades. Usar um conjunto consistente de chaves de tags facilita para você gerenciar seus recursos. Você pode pesquisar e filtrar os recursos de acordo com as tags que adicionar.

As tags não têm significado semântico no Amazon ECR e são interpretadas estritamente como uma string de caracteres. Além disso, as tags não são automaticamente atribuídas aos seus recursos. Você pode editar chaves de tags e valores, e você pode remover as tags de um recurso a qualquer momento. Você pode 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.

Pode trabalhar com etiquetas utilizando o Console de gerenciamento da AWS, o AWS CLIe o Amazon ECR API.

Se estiver a utilizar AWS Identity and Access Management (IAM), pode controlar quais os utilizadores no seu AWS conta tem permissão para criar, editar ou eliminar etiquetas.

Marcar os recursos do

Pode etiquetar novo ou existente Amazon ECR repositórios.

Se estiver a utilizar o Amazon ECR consola, pode aplicar etiquetas a novos recursos quando são criados ou recursos existentes utilizando o Etiquetas no painel de navegação a qualquer momento.

Se estiver a utilizar o Amazon ECR API, o AWS CLI, ou um AWS O SDK pode aplicar tags a novos repositórios utilizando o `tags` parâmetro na acção API do repositório de criatelidade ou utilizar o `TagResource` Ação API para aplicar tags aos recursos existentes. Para mais informações, consulte tagresource.

Além disso, se as tags não puderem ser aplicadas durante a criação do repositório, nós reverteremos o processo de criação do repositório. Isso garante que os repositórios sejam criados com tags ou, então, não criados, e que nenhum repositório seja deixado sem tags. Ao marcar com tags os repositórios no momento da criação, você elimina a necessidade de executar scripts personalizados de uso de tags após a criação do repositório.

Restrições de tags

As restrições básicas a seguir se aplicam às tags.

- Número máximo de tags por repositório – 50
- Em todos os repositórios, cada chave de tag deve ser exclusiva e pode ter apenas um valor.
- Comprimento máximo da chave – 128 caracteres Unicode em UTF-8
- Valor máximo da chave – 256 caracteres Unicode em UTF-8
- Se seu esquema de marcação for usado em vários serviços e recursos da , lembre-se de que outros serviços podem ter restrições nos caracteres permitidos. Em geral, os caracteres permitidos são: letras, números e espaços representáveis em UTF-8 e os seguintes caracteres: + - = . _ : / @.
- As chaves e os valores de tags diferenciam maiúsculas de minúsculas.
- Não utilize o `aws:` prefixo para teclas ou valores; está reservado para AWS utilizar. Você não pode editar nem excluir chaves nem valores de tag com esse prefixo. As tags com esse prefixo não contam para as tags por limite de recurso.

Marcar recursos para faturamento

As tags que você adiciona aos repositórios do Amazon ECR são úteis ao analisar a alocação de custos depois de habilitá-las em seu Relatório de custo e uso. Para obter mais informações, consulte Relatórios de uso do Amazon ECR (p. 90).

Para ver o custo dos recursos combinados, você pode organizar as informações de faturamento com base nos recursos com os mesmos valores da chave da tag. Por exemplo, você pode 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 mais informações sobre a configuração de um relatório de alocação de custos com etiquetas, consulte O relatório de alocação de custos mensais no Guia do usuário do AWS Billing and Cost Management.

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

Trabalhar com tags usando o console

Com o console do Amazon ECR, é possível gerenciar as tags associadas aos repositórios novos ou existentes.

Quando seleccionar um repositório específico no Amazon ECR consola, pode ver as etiquetas selecionando Etiquetas no painel de navegação.

Para adicionar uma tag a um repositório

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 Repositórios.
4. No Repositórios , escolha o repositório para ver.
5. No Repositórios: ***repository_name*** página, selecione Etiquetas no painel de navegação.
6. No Etiquetas página, selecione Adicionar etiquetas, Adicionar etiqueta.
7. No Editar etiquetas , especifique a chave e o valor para cada tag e, em seguida, escolha Guardar.

Para excluir uma tag de um recurso individual

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 Repositórios , escolha o repositório para ver.
4. No Repositórios: ***repository_name*** página, selecione Etiquetas no painel de navegação.
5. No Etiquetas página, selecione Editar.
6. No Editar etiquetas página, selecione Remover para cada tag que pretende eliminar e escolha Guardar.

Trabalhar com etiquetas utilizando o AWS CLI ou API

Use o seguinte para adicionar, atualizar, listar e excluir as tags para seus recursos. A documentação correspondente traz exemplos.

Suporte de etiquetagem para Amazon ECR Recursos

Tarefa	CLI DA AWS	Ação API
Adicione ou sobrescreva uma ou mais tags.	tag-resource	TagResource ()
Exclua uma ou mais tags.	untag-resource	UntagResource

Os exemplos a seguir mostram como gerenciar tags usando a AWS CLI.

Exemplo: = 1. Marcar um repositório existente

O comando a seguir marca um repositório existente.

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

Exemplo: = 2. Etiquetar um repositório existente com várias etiquetas

O comando a seguir marca um repositório existente.

```
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 Desmarcar um repositório existente

O comando a seguir exclui uma tag de um repositório existente.

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

Exemplo 4 Listar etiquetas para um repositório

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

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

Exemplo: = 5 Crie um repositório e aplique uma etiqueta

O seguinte comando cria um repositório nomeado `test-repo` e adiciona uma etiqueta com chave `team` e valor `devs`.

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

Images

Amazon Elastic Container Registry (Amazon ECR) armazena imagens Docker, imagens Open Container Initiative (OCI) e artefactos compatíveis com OCI em repositórios. Pode utilizar o CLI do Docker ou o seu cliente preferido para enviar e extrair imagens de e para os seus repositórios.

> **Important**
>
> O Amazon ECR exige que os usuários concedam permissões para a API `ecr:GetAuthorizationToken` por meio de uma política do IAM antes que eles possam fazer a autenticação em um registro e enviar ou extrair qualquer imagem de um repositório do Amazon ECR. O Amazon ECR fornece várias políticas gerenciadas do IAM para controlar o acesso de usuários em níveis variados. Para obter mais informações, consulte Amazon Elastic Container RegistryExemplos de políticas baseadas em identidade do (p. 69).

Tópicos

Enviar uma imagem

Pode enviar as suas imagens Docker para um Amazon ECR repositório com o docker push comando.

> **Important**
>
> O Amazon ECR exige que os usuários concedam permissões para a API `ecr:GetAuthorizationToken` por meio de uma política do IAM antes que eles possam fazer a autenticação em um registro e enviar ou extrair qualquer imagem de um repositório do Amazon ECR. O Amazon ECR fornece várias políticas gerenciadas do IAM para controlar o acesso de usuários em níveis variados. Para obter mais informações, consulte Amazon Elastic Container RegistryExemplos de políticas baseadas em identidade do (p. 69).

O Amazon ECR também oferece suporte à criação e ao envio de listas de manifestos do Docker que são usadas para imagens multiarquitetura. Cada imagem referenciada em uma lista de manifestos já

deve ter sido enviada para seu repositório. Para obter mais informações, consulte Enviar uma imagem multiarquitetura (p. 31).

Para enviar uma imagem de Docker a 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 obter mais informações, consulte Autenticação do registo (p. 14).
2. Se o seu repositório de imagens não existir no registro que você pretende enviar, crie-o. Para obter mais informações, consulte Criar um repositório (p. 17).
3. Identifique a imagem a ser enviada. Execute o comando docker images para listar as imagens em seu sistema.

   ```
   docker images
   ```

   Pode identificar uma imagem com o *repository:tag* ou a 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 tags da imagem para usar. O formato do registro é *aws_account_id*.dkr.ecr.*region*.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 seguinte identifica uma imagem com a ID *e9ae3c220b23* como *aws_account_id*.dkr.ecr.*region*.amazonaws.com/my-web-app

   ```
   docker tag e9ae3c220b23 aws_account_id.dkr.ecr.region.amazonaws.com/my-web-app
   ```
5. Envie a imagem usando o comando docker push:

   ```
   docker push aws_account_id.dkr.ecr.region.amazonaws.com/my-web-app
   ```
6. (Opcional) Aplique tags adicionais à sua imagem, enviando-as ao Amazon ECR repetindo Step 4 (p. 31) e Step 5 (p. 31). Você pode aplicar até 100 tags por imagem no Amazon ECR.

Enviar uma imagem multiarquitetura

O Amazon ECR oferece suporte à criação e envio de listas de manifestos do Docker usadas para imagens multiarquitetura. Uma lista de manifestos é uma lista de imagens criada com a especificação de um ou mais nomes de imagem. Normalmente, a lista de manifestos é criada de imagens que exercem a mesma função para sistemas operacionais ou arquiteturas diferentes, mas isso não é necessário. Para obter mais informações, consulte manifesto do docker.

> **Important**
>
> A CLI do Docker deve ter recursos experimentais habilitados para usar esse recurso. Para obter mais informações, consulte Recursos experimentais.

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.

As etapas a seguir podem ser usadas para criar e enviar uma lista de manifestos do Docker para um repositório do Amazon ECR. Você já deve ter as imagens enviadas ao repositório para fazer referência

no manifesto do Docker. Para obter informações sobre como enviar uma imagem, consulte Enviar uma imagem (p. 30).

Como enviar uma imagem 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 obter mais informações, consulte Autenticação do registo (p. 14).
2. Listar as imagens no repositório, confirmando as tags de imagem.

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

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.region.amazonaws.com/my-
web-app aws_account_id.dkr.ecr.region.amazonaws.com/my-web-
app:image_one_tag aws_account_id.dkr.ecr.region.amazonaws.com/my-web-app: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.region.amazonaws.com/my-web-app
```

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

```
docker manifest push aws_account_id.dkr.ecr.region.amazonaws.com/my-web-app
```

Empurrar um gráfico Helm

Amazon ECR suporta a introdução de artefactos da Open Container Initiative (OCI) nos seus repositórios. Para ver esta funcionalidade, utilize os seguintes passos para empurrar um gráfico Helm para Amazon ECR.

Para mais informações sobre como utilizar o seu Amazon ECR gráficos Helm alojados com Amazon EKS, consulte Instalar um gráfico Helm alojado em Amazon ECR com Amazon EKS (p. 57).

Para empurrar um gráfico Helm para um Amazon ECR repositório

1. Instale o cliente Helm versão 3. Para obter mais informações, consulte Instalação do leme.
2. Permitir o apoio da OCI no cliente Helm 3.

```
export HELM_EXPERIMENTAL_OCI=1
```

3. Crie um repositório para armazenar o seu gráfico Helm. Para obter mais informações, consulte Criar um repositório (p. 17).

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

4. Autentique o seu cliente Helm no Amazon ECR para o qual pretende enviar o seu gráfico Helm. Os tokens de autenticação devem ser obtidos para cada registro usado e são válidos por 12 horas. Para obter mais informações, consulte Autenticação do registo (p. 14).

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

5. Siga os passos seguintes para criar um gráfico de teste Helm. Para obter mais informações, consulte Helm Docs - Introdução.

   a. Criar um diretório com o nome `helm-tutorial` trabalhar em.

   ```
   mkdir helm-tutorial
   cd helm-tutorial
   ```

   b. Crie um gráfico Helm com o nome `mychart` e limpar o conteúdo do `templates` diretoria.

   ```
   helm create mychart
   rm -rf ./mychart/templates/*
   ```

   c. Crie um ConfigMap na `templates` pasta.

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

6. Guarde o gráfico localmente e crie um alias para o gráfico com o seu registo URI.

```
cd ..
helm chart save . mychart
helm chart save . aws_account_id.dkr.ecr.us-west-2.amazonaws.com/artifact-test:mychart
```

7. Identifique o gráfico Helm a empurrar. Execute o helm chart list para listar os gráficos Helm no seu sistema.

```
helm chart list
```

   A saída deve ser semelhante a isto:

```
REF                                                             NAME    VERSION DIGEST
 SIZE    CREATED
aws_account_id.dkr.ecr.us-west-2.amazonaws.com/artifact-tes..   mychart 0.1.0   30e0a03
 3.6 KiB 14 seconds
mychart                                                         mychart 0.1.0   ba3e62a 3.6
 KiB About a minute
```

8. Empurre o gráfico Helm usando o helm chart push comando:

```
helm chart push aws_account_id.dkr.ecr.region.amazonaws.com/artifact-test:mychart
```

9. Descreva o seu gráfico Helm.

```
aws ecr describe-images \
     --repository-name artifact-test \
```

```
    --region us-west-2
```

Na saída, verifique a `artifactMediaType` indica o tipo de artefacto adequado.

```
{
    "imageDetails": [
        {
            "registryId": "aws_account_id",
            "repositoryName": "artifact-test",
            "imageDigest":
 "sha256:f23ab9dc0fda33175e465bd694a5f4cade93eaf62715fa9390d9fEXAMPLE",
            "imageTags": [
                "mychart"
            ],
            "imageSizeInBytes": 3714,
            "imagePushedAt": 1597433021.0,
            "imageManifestMediaType": "application/vnd.oci.image.manifest.v1+json",
            "artifactMediaType": "application/vnd.cncf.helm.config.v1+json"
        }
    ]
}
```

Extrair uma imagem

Se quiser executar uma imagem de Docker que está disponível no Amazon ECR, você pode extrai-la em seu ambiente local com o comando docker pull. Você pode fazer isso com seu registro padrão ou de um registro associado com outra conta da AWS. Para usar uma imagem do Amazon ECR em uma definição de tarefas do Amazon ECS, consulte Usar imagens do Amazon ECR com o Amazon ECS (p. 55).

> Important
>
> O Amazon ECR exige que os usuários concedam permissões para a API `ecr:GetAuthorizationToken` por meio de uma política do IAM antes que eles possam fazer a autenticação em um registro e enviar ou extrair qualquer imagem de um repositório do Amazon ECR. O Amazon ECR fornece várias políticas gerenciadas do IAM para controlar o acesso de usuários em níveis variados. Para obter mais informações, consulte Amazon Elastic Container RegistryExemplos de políticas baseadas em identidade do (p. 69).

Para extrair uma imagem de Docker de um repositório no 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 obter mais informações, consulte Autenticação do registo (p. 14).
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 você receber um erro *repository-url* `not found: does not exist or no pull access`, poderá ser necessário autenticar seu cliente do Docker com o Amazon ECR. Para obter mais informações, consulte Autenticação do registo (p. 14).

Excluir uma imagem

Se você já terminou de usar uma imagem, pode excluí-la do repositório. É possível excluir uma imagem usando o Console de gerenciamento da AWS ou a AWS CLI.

Note

Se você já terminou de usar um repositório, pode excluir o repositório inteiro e todas as imagens contidas nele. Para obter mais informações, consulte Excluir um repositório (p. 20).

Para excluir uma imagem com o Console de gerenciamento da AWS

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. Em Repositórios: ***repository_name*** , selecione a caixa à esquerda da imagem a eliminar e escolha Eliminar.
6. Na caixa de diálogo Excluir imagem(ns), verifique se as imagens selecionadas devem ser excluídas e escolha Excluir.

Para excluir uma imagem com o AWS CLI

1. Liste as imagens em seu repositório de modo que você possa identificá-las pela tag ou resumo da imagem.

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

2. (Opcional) Exclua quaisquer tags indesejáveis para a imagem especificando a tag da imagem que você deseja excluir.

   Note

   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=latest
```

3. Exclua a imagem especificando o resumo da imagem a ser excluída.

   Note

   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=sha256:4f70ef7a4d29e8c0c302b13e25962d8f7a0bd304c7c2c1a9d6fa3e9de6bf552d
```

Remarcar uma imagem

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 de imagem a fim de que a imagem remarque-o e grave-o em uma variável de ambiente. Neste exemplo, o manifesto de uma imagem com a etiqueta, *latest*, no repositório, *amazonlinux*é escrita na variável de ambiente, *MANIFEST*.

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

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

   > **Note**
   >
   > Se a opção `--image-tag` não estiver disponível na sua versão da AWS CLI, atualize para a versão mais recente. Para obter mais informações, consulte Instalar a interface de linha de comando da AWS no Guia do usuário do 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. No resultado a seguir, a imagem têm as tags `latest` e `2017.03`.

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

   Resultado:

   ```
   {
       "imageDetails": [
           {
               "imageSizeInBytes": 98755613,
               "imageDigest":
    "sha256:8d00af8f076eb15a33019c2a3e7f1f655375681c4e5be157a2685dfe6f247227",
               "imageTags": [
                   "latest",
                   "2017.03"
               ],
               "registryId": "aws_account_id",
               "repositoryName": "amazonlinux",
               "imagePushedAt": 1499287667.0
           }
       ]
   ```

```
}
```

Como remarcar uma imagem (AWS Tools para Windows PowerShell)

Para remarcar uma imagem com a AWS Tools para 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 etiqueta, *latest*, no repositório, *amazonlinux*é escrita na variável de ambiente, *$Image*.

   Note

   Se você não tiver o cmdlet Get-ECRImageBatch disponível no sistema, consulte Configurar do AWS Tools para Windows PowerShell no Guia do usuário do AWS Tools para Windows PowerShell.

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

2. Escreva o manifesto da imagem no *$Manifest* variável de ambiente.

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

3. Use a opção `-ImageTag` do cmdlet Write-ECRImage para colocar o manifesto de 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. No resultado a seguir, a imagem têm as tags `latest` e `2017.09`.

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

   Resultado:

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

Políticas de ciclo de vida

Amazon ECRAs políticas de ciclo de vida do permitem que você especifique o gerenciamento do ciclo de vida das imagens em um repositório. Uma política de ciclo de vida é um conjunto de uma ou mais regras em que cada regra define uma ação do Amazon ECR. As ações aplicam-se a imagens que contêm tags prefixadas de determinadas strings. Isso permite a automação da limpeza de imagens não usadas, por exemplo imagens prestes a expirar com base em seu tempo de vida ou contagem. Após criar uma política de ciclo de vida, você deve esperar que as imagens afetadas expirem em 24 horas.

Tópicos

Modelo da apólice 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. Para ver exemplos de política do ciclo de vida, consulte Exemplos de políticas do ciclo de vida (p. 43).

```
{
    "rules": [
        {
            "rulePriority": integer,
            "description": "string",
            "selection": {
                "tagStatus": "tagged"|"untagged"|"any",
                "tagPrefixList": list<string>,
                "countType": "imageCountMoreThan"|"sinceImagePushed",
                "countUnit": "string",
                "countNumber": integer
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

Note

O `tagPrefixList` parâmetro só é utilizado se `tagStatus` é `tagged`. O `countUnit` parâmetro só é utilizado se `countType` é `sinceImagePushed`. O `countNumber` parâmetro só é utilizado se `countType` está definido para `imageCountMoreThan`.

Parâmetros da política do ciclo de vida

As políticas de ciclo de vida são divididas nas partes a seguir:

Tópicos

Prioridade da regra

`rulePriority`

Tipo: inteiro.

Obrigatório: sim

Define a ordem em que as regras são avaliadas, da menor para a maior. Uma regra da política do ciclo de vida com uma prioridade de `1` será atuado primeiro, uma regra com prioridade de `2` será seguinte, e por isso em. Quando adiciona regras a uma política do ciclo de vida, tem de dar-lhes um valor único para `rulePriority`. Os valores não precisam de ser sequenciais em todas as regras de uma apólice. Uma regra com um `tagStatus` valor de `any` tem de ter o valor mais elevado para `rulePriority` e ser avaliado em último lugar.

Description

`description`

Tipo: string.

Obrigatório: não

(Opcional) Descreve a finalidade de uma regra em uma política de ciclo de vida.

Estado da etiqueta

`tagStatus`

Tipo: string.

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 especificar `any`, então todas as imagens têm a regra que lhes foi aplicada. Se especificar `tagged`, então tem também de especificar uma `tagPrefixList` valor. Se você especificar `untagged`, você deverá omitir `tagPrefixList`.

Lista de prefixo de etiquetas

`tagPrefixList`

Tipo: list[string]

Exigido: sim, somente se `tagStatus` for definido como tagged

Apenas utilizado se especificado `"tagStatus": "tagged"`. Tem de especificar uma lista de prefixos de etiquetas de imagem separadas por vírgulas para tomar medidas com a sua política de ciclo de vida. Por exemplo, se as suas imagens forem marcadas como `prod`, `prod1`, `prod2`, etc., utilizaria o prefixo da etiqueta `prod` para especificar todos os. Se você especificar várias tags, apenas imagens com todas as tags especificadas serão selecionadas.

Tipo de contagem

`countType`

Tipo: string.

Obrigatório: sim

Especifique um tipo de contagem a ser aplicado às imagens.

Se `countType` está definido para `imageCountMoreThan`, também especifica `countNumber` para criar uma regra que defina um limite no número de imagens que existem no seu repositório. Se `countType` está definido para `sinceImagePushed`, também especifica `countUnit` e `countNumber` para especificar um limite de tempo nas imagens que existem no seu repositório.

Unidade de contagem

`countUnit`

Tipo: string.

Exigido: sim, somente se `countType` for definido como `sinceImagePushed`

Especifique uma unidade de contagem de `days` para indicar que como a unidade de tempo, para além de `countNumber`, que é o número de dias.

Isto só deve ser especificado quando `countType` é `sinceImagePushed` um erro ocorre se especificar uma unidade de contagem quando `countType` 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` é `imageCountMoreThan`, então o valor é o número máximo de imagens que pretende reter no seu repositório. Se o `countType` é `sinceImagePushed`, então o valor é o limite de idade máximo para as suas imagens.

Action

`type`

Tipo: string.

Obrigatório: sim

Especifique um tipo de ação. O valor suportado é `expire`.

Regras de avaliação da política do ciclo de vida

O avaliador da política de ciclo de vida é responsável por analisar o JSON em formato de texto simples e aplicá-lo a imagens no repositório especificado. As seguintes regras devem ser notadas ao criar uma política de ciclo de vida:

- 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.
- A expiração é sempre solicitada por `pushed_at_time` e expira sempre as imagens mais antigas antes das mais novas.
- Quando utilizar o `tagPrefixList`, uma imagem corresponde com sucesso se todos das etiquetas no `tagPrefixList` o valor é correspondente a qualquer uma das etiquetas da imagem.
- Com `countType = imageCountMoreThan`, as imagens são ordenadas do mais novo para o mais antigo com base em `pushed_at_time` e depois todas as imagens superiores à contagem especificada expiraram.
- Com `countType = sinceImagePushed`, todas as imagens que `pushed_at_time` é anterior ao número de dias especificado com base em `countNumber` expirou.

Criar uma pré-visualização da política do ciclo de vida

Uma visualização da política de ciclo de vida permite que você veja o impacto de uma política de ciclo de vida em um repositório de imagens antes que você a execute. O procedimento a seguir mostra como criar uma visualização de política de ciclo de vida.

Para criar uma visualização de política de ciclo de vida usando o 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, escolha Repositórios e selecione um repositório.
4. No Repositórios: ***`repository_name`*** , no painel de navegação escolha Política do ciclo de vida.
5. No Repositórios: ***`repository_name`***: Política do ciclo de vida página, escolher Editar regras de teste, Criar regra.
6. Insira os seguintes detalhes para sua a regra da política de ciclo de vida:
   a. Para Prioridade da regra, escreva um número para a prioridade da regra.
   b. Para Descrição da regra, escreva uma descrição para a regra da política do ciclo de vida.
   c. Para Estado da imagem, escolha Marcado, Sem etiqueta, ou Qualquer.
   d. Se especificou `Tagged` para Estado da imagem, então para Prefixos de etiquetas, pode opcionalmente especificar uma lista de etiquetas de imagens em que deve agir com a sua política do ciclo de vida. Se você especificou `Untagged`, este campo deve estar vazio.
   e. Para Critérios de correspondência, escolha valores para Como a imagem foi premida ou Contagem de imagens mais do que (se aplicável).
7. Escolher Guardar.

8. Crie regras de política de ciclo de vida adicionais repetindo as etapas de 5 a 7.
9. Para executar a pré-visualização da política do ciclo de vida, escolha Guardar e executar teste.
10. Abaixo Correspondências de imagens para as regras do ciclo de vida do teste, reveja o impacto da pré-visualização da política do seu ciclo de vida.
11. Se estiver satisfeito com os resultados da pré-visualização, escolha Aplicar como política do ciclo de vida para criar uma política do ciclo de vida com as regras especificadas.

Note

Após criar uma política de ciclo de vida, você deve esperar que as imagens afetadas expirem em 24 horas.

Criar uma política de ciclo de vida

Uma política de ciclo de vida permite que você crie um conjunto de regras que expira imagens de repositório não utilizadas. O procedimento a seguir mostra como criar uma política de ciclo de vida. Após criar uma política de ciclo de vida, você deve esperar que as imagens afetadas expirem em 24 horas.

Como criar uma política de ciclo de vidaAWS CLI)

Para criar uma política do ciclo de vida utilizando o AWS CLI

1. Obtenha o ID do repositório para o qual a política de ciclo de vida será criada:

```
aws ecr describe-repositories
```

2. Criar uma política de ciclo de vida

```
aws ecr put-lifecycle-policy [--registry-id <string>] --repository-name <string> --
lifecycle-policy-text <string>
```

Como criar uma política de ciclo de vidaConsole de gerenciamento da AWS)

Para criar uma política de ciclo de vida usando o 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, escolha Repositórios e selecione um repositório.
4. No Repositórios: ***repository_name*** , no painel de navegação escolha Política do ciclo de vida.
5. No Repositórios: ***repository_name***: Política do ciclo de vida página, escolher Criar regra.
6. Insira os seguintes detalhes para sua a regra da política de ciclo de vida:
   a. Para Prioridade da regra, escreva um número para a prioridade da regra.
   b. Para Descrição da regra, escreva uma descrição para a regra da política do ciclo de vida.
   c. Para Estado da imagem, escolha Marcado, Sem etiqueta, ou Qualquer.
   d. Se especificou `Tagged` para Estado da imagem, então para Prefixos de etiquetas, pode opcionalmente especificar uma lista de etiquetas de imagens em que deve agir com a sua política do ciclo de vida. Se você especificou `Untagged`, este campo deve estar vazio.
   e. Para Critérios de correspondência, escolha valores para Como a imagem foi premida ou Contagem de imagens mais do que (se aplicável).

7. Escolher Guardar.

Exemplos de políticas do ciclo de vida

Veja a seguir exemplos de políticas de ciclo de vida, mostrando a sintaxe.

Tópicos

Filtrar na idade da imagem

O exemplo a seguir mostra a sintaxe de política de ciclo de vida para uma política que expira imagens não marcadas com mais de `14` dias:

```
{
    "rules": [
        {
            "rulePriority": 1,
            "description": "Expire images older than 14 days",
            "selection": {
                "tagStatus": "untagged",
                "countType": "sinceImagePushed",
                "countUnit": "days",
                "countNumber": 14
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

Filtrar na contagem de imagens

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 em 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",
                "tagPrefixList": ["prod"],
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        },
        {
            "rulePriority": 2,
            "description": "Rule 2",
            "selection": {
                "tagStatus": "tagged",
                "tagPrefixList": ["beta"],
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

A lógica dessa política de ciclo de vida seria:

- Regra 1 identifica imagens marcadas com prefixo `prod`. Deve marcar imagens, começando pelo mais antigo, até que haja uma ou menos imagens que correspondam. Ela marca a imagem A para expiração.
- Regra 2 identifica imagens marcadas com prefixo `beta`. Deve marcar imagens, começando pelo mais antigo, até que haja uma ou menos imagens que correspondam. 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 expirou.

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",
                "tagPrefixList": ["beta"],
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        },
        {
            "rulePriority": 2,
            "description": "Rule 2",
            "selection": {
                "tagStatus": "tagged",
                "tagPrefixList": ["prod"],
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

A lógica dessa política de ciclo de vida seria:

- Regra 1 identifica imagens marcadas com `beta`. Deve marcar imagens, começando pelo mais antigo, até que haja uma ou menos imagens que correspondam. Ela vê todas as três imagens e marcaria as imagens A e B para expiração.
- Regra 2 identifica imagens marcadas com `prod`. Deve marcar imagens, começando pelo mais antigo, até que haja uma ou menos imagens que correspondam. 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 expiraram.

Filtrar em várias etiquetas numa única regra

Os seguintes exemplos especificam a sintaxe de política de ciclo de vida para vários prefixos de tag 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 prefixos de tags são especificados em uma única regra, as imagens devem corresponder a todos os prefixos de tag 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",
                "tagPrefixList": ["alpha", "beta"],
                "countType": "sinceImagePushed",
                "countNumber": 5,
                "countUnit": "days"
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

A lógica dessa política de ciclo de vida seria:

- Regra 1 identifica imagens marcadas com `alpha` e `beta`. Vê imagens C e F. Deve marcar imagens com mais de cinco dias, que seriam Imagens C.
- Resultado: A imagem C expirou.

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",
                "tagPrefixList": ["alpha", "beta"],
                "countType": "imageCountMoreThan",
                "countNumber": 1
            },
            "action": {
                "type": "expire"
            }
        }
    ]
}
```

A lógica dessa política de ciclo de vida seria:

- Regra 1 identifica imagens marcadas com `alpha` e `beta`. 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 expiraram.

Filtrar em 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, Lista: [], Premido: 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 expiraram.

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, Lista: [], Premido: 3 dias atrás
- Imagem C, Taglist: ["alpha-2"], Enviada: 2 dias atrás
- Imagem D, Taglist: ["git hash"], Enviada: 1 dia atrás
- Imagem E, Lista: [], Premido: 1 dia atrás

```
{
    "rules": [
        {
            "rulePriority": 1,
            "description": "Rule 1",
            "selection": {
                "tagStatus": "tagged",
                "tagPrefixList": ["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:

- Regra 1 identifica imagens marcadas com `alpha`. Identifica imagens A e C. Deve manter a imagem mais recente e marcar o resto para expiração. E 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 expiraram.

Mutabilidade de tag de imagem

É possível configurar um repositório para ser imutável a fim de impedir que as tags de imagem sejam substituídas. Assim que o repositório estiver configurado para tags imutáveis, um erro `ImageTagAlreadyExistsException` será retornado se você tentar enviar uma imagem com uma tag que já existe no repositório.

Use o Console de gerenciamento da AWS e as ferramentas da AWS CLI a fim de definir a mutabilidade de tag de imagem para um novo repositório durante a criação ou para um repositório existente a qualquer momento. Para conhecer as etapas do console, consulte Criar um repositório (p. 17) e Editar um repositório (p. 19).

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 para Windows PowerShell)

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

Como atualizar as configurações de mutabilidade de tag de imagem para um repositório existente

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
```

- Write-ECRImageTagMutability (AWS Tools para Windows PowerShell)

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

Verificação de imagens

Amazon ECR a digitalização de imagens ajuda a identificar vulnerabilidades de software nas suas imagens de recipiente. Amazon ECR utiliza a base de dados de Vulnerabilidades e Exposições Comuns (Common Vulnerabilities and Exposures, CVE) do projeto de código aberto Clair e fornece-lhe uma lista de resultados de scan. Examine as descobertas da verificação para obter informações sobre a segurança das imagens de contêiner que estão sendo implantadas. Para mais informações sobre a Clair, consulte Detenção no GitHub.

O Amazon ECR usa a gravidade de um CVE da fonte de distribuição upstream, se disponível. Caso contrário, usamos a pontuação do Common Vulnerability Scoring System (CVSS – Sistema de pontuação de vulnerabilidade comum). 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.

É possível verificar manualmente as imagens de contêiner armazenadas no Amazon ECR, ou configurar seus repositórios para verificar imagens ao enviá-las para um repositório. As últimas descobertas da verificação de imagem concluídas podem ser recuperadas para cada imagem. O Amazon ECR envia um evento para o Amazon EventBridge (anteriormente chamado de Eventos do CloudWatch) quando uma verificação de imagem é concluída. Para obter mais informações, consulte Amazon ECR eventos e EventBridge (p. 91).

Para obter detalhes de solução de problemas para alguns problemas comuns ao digitalizar imagens, consulte Solução de problemas de verificação de imagem (p. 111).

Tópicos

Configurar um repositório para verificar ao enviar

Você pode definir as configurações de verificação de imagem para um novo repositório durante a criação ou para um repositório existente. Quando a opção scan on push (verificar ao enviar) estiver habilitada, as imagens serão verificadas depois de serem enviadas para um repositório. Se a opção scan on push (verificar ao enviar) estiver desabilitada em um repositório, você deverá iniciar manualmente cada verificação de imagem para obter os resultados da verificação.

Tópicos

Criar um repositório para verificar ao enviar

Quando um novo repositório é configurado para scan on push (verificar ao enviar), todas as novas imagens enviadas para o repositório serão verificadas. Depois, os resultados da última verificação de imagem

concluída poderão ser recuperados. Para obter mais informações, consulte Recuperar descobertas de verificação de imagem (p. 53).

Para obter as etapas do Console de gerenciamento da AWS, consulte Criar um repositório (p. 17).

Como criar um repositório configurado para verificar ao enviar (AWS CLI)

Use o comando a seguir para criar um novo repositório com a opção scan on push (verificar ao enviar) configurada para a imagem.

- create-repository (AWS CLI)

```
aws ecr create-repository --repository-name name --image-scanning-configuration
 scanOnPush=true --region us-east-2
```

Como criar um repositório configurado para verificar ao enviar (AWS Tools para Windows PowerShell)

Use o comando a seguir para criar um novo repositório com a opção scan on push (verificar ao enviar) configurada para a imagem.

- New-ECRRepository (AWS Tools para Windows PowerShell)

```
New-ECRRepository -RepositoryName name -ImageScanningConfiguration_ScanOnPush true -
Region us-east-2 -Force
```

Configurar um repositório existente para verificar ao enviar

Os repositórios existentes podem ser configurados para verificar as imagens quando você as enviar para um repositório. Essa configuração se aplicará a futuros envios de imagens. Depois, os resultados da última verificação de imagem concluída poderão ser recuperados. Para obter mais informações, consulte Recuperar descobertas de verificação de imagem (p. 53).

Para obter as etapas do Console de gerenciamento da AWS, consulte Editar um repositório (p. 19).

Como editar as configurações de um repositório existente (AWS CLI)

Use o comando a seguir para editar as configurações de verificação de imagem de um repositório existente.

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

```
aws ecr put-image-scanning-configuration --repository-name name --image-scanning-
configuration scanOnPush=true --region us-east-2
```

> **Note**
>
> Para desativar a opção scan on push (verificar ao enviar) de imagens para um repositório, especifique `scanOnPush=false`.

Como editar as configurações de um repositório existente (AWS Tools para Windows PowerShell)

Use o comando a seguir para editar as configurações de verificação de imagem de um repositório existente.

- New-ECRRepository (AWS Tools para Windows PowerShell)

```
Write-ECRImageScanningConfiguration -RepositoryName name -
ImageScanningConfiguration_ScanOnPush true -Region us-east-2 -Force
```

Verificar manualmente uma imagem

Você pode iniciar verificações de imagens manualmente quando quiser verificar imagens em repositórios que não estejam configuradas para scan on push (verificar ao enviar). Uma imagem só pode ser verificada uma vez por dia. Esse limite inclui a opção scan on push (verificar ao enviar) inicial, se habilitada, e quaisquer verificações manuais.

Para obter detalhes de solução de problemas para alguns problemas comuns ao digitalizar imagens, consulte Solução de problemas de verificação de imagem (p. 111).

Como iniciar a verificação manual de uma imagem (console)

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

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).

Como iniciar a verificação manual de uma imagem (AWS CLI)

Use o comando da AWS CLI a seguir para iniciar a verificação manual de uma imagem. É possível especificar uma imagem usando a `imageTag` ou o `imageDigest`. Ambos podem ser obtidos usando o comando list-images da 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
```

Como iniciar a verificação manual de uma imagem (AWS Tools para Windows PowerShell)

Use o comando da AWS Tools para Windows PowerShell a seguir para iniciar a verificação manual de uma imagem. É possível especificar uma imagem usando a `ImageId_ImageTag` ou o `ImageId_ImageDigest`. Ambos podem ser obtidos com o comando Get-ECRImage da CLI.

- Get-ECRImageScanFinding (AWS Tools para 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
```

Recuperar descobertas de verificação de imagem

É possível recuperar as descobertas da verificação para a última verificação de imagem concluída. As descobertas listam por gravidade as vulnerabilidades de software que foram descobertas, com base no banco de dados de vulnerabilidades e exposições comuns (CVEs).

Para obter detalhes de solução de problemas para alguns problemas comuns ao digitalizar imagens, consulte Solução de problemas de verificação de imagem (p. 111).

Como recuperar descobertas de verificação de imagem (console)

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

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.

Como recuperar descobertas da verificação de imagem (AWS CLI)

Use o comando da AWS CLI a seguir para recuperar as descobertas de verificação de imagem usando a AWS CLI. É 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
```

Como recuperar descobertas da verificação de imagem (AWS Tools para Windows PowerShell)

Use o comando do AWS Tools para Windows PowerShell a seguir para recuperar as descobertas de verificação de imagem. É possível especificar uma imagem usando a `ImageId_ImageTag` ou o `ImageId_ImageDigest`. Ambos podem ser obtidos com o comando Get-ECRImage da CLI.

- Get-ECRImageScanFinding (AWS Tools para 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
```

Formatos de manifesto de imagem de contêiner

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 por imagem.
- Suporte para armazenar imagens de contêiner do Windows. Para obter mais informações, consulte Como enviar por push imagens do Windows para o Amazon ECR no Amazon Elastic Container Service Developer Guide.

Conversão de manifesto de imagem do Amazon ECR

Quando você envia imagens ao Amazon ECR e as extrai dele, o cliente de mecanismo de contêiner (por exemplo, Docker) se comunica com o registro para concordar com 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 de manifesto de imagem é armazenado como schema 1 de manifesto V2 de imagem de Docker. Quando você envia uma imagem ao Amazon ECR com o Docker versão 1.10 ou posterior, o formato de manifesto de imagem é armazenado como schema 2 de manifesto V2 de imagem de Docker.

Quando você extrai uma imagem do Amazon ECR por tag, o Amazon ECR retorna o formato de manifesto de imagem que é 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, Amazon ECR converte o manifesto de imagem em um formato que é entendido pelo cliente. Por exemplo, se um cliente 1.9 de Docker solicitar um manifesto de imagem armazenado como Esquema 2 do Manifesto de Imagem de docker, Amazon ECR devolve o manifesto no formato de esquema 1 V2 do manifesto de imagem de docker. A tabela a seguir descreve as conversões disponíveis compatíveis com o 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 extrai uma imagem por digestão, não há conversões disponíveis. É necessário que seu cliente entenda o formato de 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 converterá o manifesto de imagem em um formato que o cliente possa entender. A extração da imagem é bem-sucedida.

Usar imagens do Amazon ECR com o Amazon ECS

Você pode usar as imagens de contêiner hospedadas no Amazon ECR em suas definições de tarefas do Amazon ECS, mas precisa atender aos seguintes pré-requisitos.

- Ao usar o tipo de inicialização EC2 para as tarefas do Amazon ECS, suas instâncias de contêiner devem estar usando pelo menos a versão 1.7.0 do agente de contêiner do Amazon ECS. A versão mais recente da AMI otimizada para o Amazon ECS é compatível com imagens do Amazon ECR em definições de tarefas. Para obter mais informações, incluindo os IDs da AMIs mais recentes otimizadas para o Amazon ECS, consulte Versões de AMIs otimizadas para o Amazon ECS no Amazon Elastic Container Service Developer Guide.
- A função do IAM da instância de contêiner do Amazon ECS (`ecsInstanceRole`) usada deve conter as seguintes permissões da política do IAM para o Amazon ECR.

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

Se você usar a política gerenciada AmazonEC2ContainerServiceforEC2Role sua IAM função de instância de contêiner terá as permissões adequadas. Para verificar se sua função é compatível com o Amazon ECR, consulte Função do IAM de instância de contêiner do Amazon ECS no Amazon Elastic Container Service Developer Guide.

- Em suas definições de tarefas do Amazon ECS, 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-web-app*`:`*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-web-app:latest",
            ...
        }
    ],
    ...
}
```

Usar imagens do Amazon ECR com o Amazon EKS

Você pode usar suas imagens do Amazon ECR com o Amazon EKS, mas precisa atender aos seguintes pré-requisitos:

- A função do IAM do nó de operador do Amazon EKS (`NodeInstanceRole`) usada com os nós de operador deve ter 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` ou os modelos do AWS CloudFormation em Conceitos básicos do Amazon EKS para criar seu cluster e seus grupos de nó de operador, essas permissões do IAM serão aplicadas à função do IAM do nó de trabalho por padrão.

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

Instalar um gráfico Helm alojado em Amazon ECR com Amazon EKS

Os seus gráficos Helm alojados em Amazon ECR pode ser instalado no seu Amazon EKS conjuntos de amostras. Os passos seguintes demonstram este.

Prerequisites

Antes de começar, certifique-se de que os seguintes passos foram concluídos.

- Instale o cliente Helm versão 3. Para obter mais informações, consulte Instalação do leme.
- Empurrou um gráfico Helm para o seu Amazon ECR repositório. Para obter mais informações, consulte Empurrar um gráfico Helm (p. 32).
- Você configurou `kubectl` trabalhar com Amazon EKS. Para obter mais informações, consulte Criar uma `kubeconfig` por Amazon EKS na Amazon EKS Guia do usuário. Se os seguintes comandos do seu cluster forem bem-sucedidos, está corretamente configurado.

```
kubectl get svc
```

Instalar um Amazon ECR alojado num gráfico Helm Amazon EKS cluster

1. Permitir o apoio da OCI no cliente Helm 3.

   ```
   export HELM_EXPERIMENTAL_OCI=1
   ```

2. Autentique o seu cliente Helm no Amazon ECR registo de que o seu gráfico Helm está alojado. Os tokens de autenticação devem ser obtidos para cada registro usado e são válidos por 12 horas. Para obter mais informações, consulte Autenticação do registo (p. 14).

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

3. Puxe o seu gráfico Helm para a sua cache local.

   ```
   helm chart pull aws_account_id.dkr.ecr.region.amazonaws.com/repository-name:mychart
   ```

4. Exporte o gráfico para um diretório local. Neste exemplo, usamos um diretório chamado `charts`.

   ```
   helm chart export aws_account_id.dkr.ecr.region.amazonaws.com/repository-name:mychart
    --destination ./charts
   ```

5. Instale o gráfico.

   ```
   helm install ecr-chart-demo ./mychart
   ```

   A saída deve ser semelhante a isto:

   ```
   NAME: ecr-chart-demo
   LAST DEPLOYED: Wed Sep  2 14:32:07 2020
   NAMESPACE: default
   STATUS: deployed
   REVISION: 1
   ```

```
NOTES:
```

6. Verifique a instalação do gráfico. A saída será uma representação YAML dos recursos Kubernetes implementados pelo gráfico.

```
helm get manifest ecr-chart-demo
```

7. (Opcional) Consulte o seu gráfico Helm a correr no seu Amazon EKS espaço.

```
kubectl get pods --all-namespaces
```

8. Quando terminar, pode remover a libertação do gráfico do seu cluster.

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

Imagem de contêiner do Amazon Linux

A imagem do contêiner Amazon Linux é criada a partir dos mesmos componentes de software que são incluídos no AMI de Amazon Linux. Está disponível para uso no ambiente como uma imagem de base para cargas de trabalho do Docker. Se você já usa a AMI do Amazon Linux para aplicativos no Amazon EC2, pode colocar facilmente seus aplicativos em contêineres com a imagem de contêiner do Amazon Linux.

É possível usar a imagem de contêiner do Amazon Linux em seu ambiente de desenvolvimento local e enviar seu aplicativo à nuvem AWS usando o Amazon ECS. Para obter mais informações, consulte Usar imagens do Amazon ECR com o Amazon ECS (p. 55).

A imagem de contêiner do Amazon Linux está disponível no Amazon ECR e no Docker Hub. O suporte para a imagem de contêiner do Amazon Linux pode ser encontrado nos fóruns de desenvolvedores da AWS.

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

1. Autentique o cliente do Docker para o registro do Amazon ECR da imagem de contêiner do Amazon Linux. Os tokens de autenticação são válidos por 12 horas. Para obter mais informações, consulte Autenticação do registo (p. 14).

   Note

   lá estão get-login-password está disponível na caixa de verificação AWS CLI começando pela versão `1.17.10`. Para obter mais informações, consulte Instalação da Interface da Linha de Comando AWS na Guia do usuário do AWS Command Line Interface.

```
aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-
stdin 137112412989.dkr.ecr.us-east-1.amazonaws.com
```

   Resultado:

```
Login succeeded
```

   Important

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

2. (Opcional) Você pode listar as imagens em um repositório do Amazon Linux com o comando aws ecr list-images. A tag `latest` sempre corresponde à imagem mais recente de contêiner do Amazon Linux que está disponível.

```
aws ecr list-images --region us-east-1 --registry-id 137112412989 --repository-name
 amazonlinux
```

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

```
docker pull 137112412989.dkr.ecr.us-east-1.amazonaws.com/amazonlinux:latest
```

4. (Opcional) Execute o contêiner localmente.

```
docker run -it 137112412989.dkr.ecr.us-east-1.amazonaws.com/amazonlinux:latest /bin/
bash
```

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

1. Extraia a imagem de 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
```

Segurança em Amazon Elastic Container Registry

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

A segurança é uma responsabilidade compartilhada entre a AWS e você. O modelo de responsabilidade partilhada descreve isto como segurança de a nuvem e a segurança em a nuvem:

- Segurança da nuvem – AWS é responsável por proteger a infraestrutura que funciona AWS serviços no AWS Nuvem. AWS também lhe fornece serviços que pode utilizar com segurança. Os auditores terceiros testam e verificam regularmente a eficácia da nossa segurança como parte do AWS programas de conformidade. Para saber mais sobre os programas de conformidade que se aplicam a Amazon ECR, consulte AWS Serviços no Âmbito pelo Programa de Conformidade.
- Segurança na nuvem – A sua responsabilidade é determinada pelo AWS que utiliza. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da sua 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 de conformidade. Também aprende a utilizar outros serviços AWS que o ajudam a monitorizar e a proteger o seu Amazon ECR recursos.

Tópicos

Identity and Access Management para o Amazon Elastic Container Registry

O AWS Identity and Access Management (IAM) é um serviço da AWS que ajuda um administrador a controlar com segurança o acesso aos recursos da AWS. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (ter permissões) para usar os recursos do Amazon ECR. O IAM é um serviço da AWS que pode ser usado sem custo adicional.

Tópicos

- Resolução de problemas Amazon Elastic Container Registry Identidade e acesso (p. 73)

Audience

O uso do AWS Identity and Access Management (IAM) varia, dependendo do trabalho que você realiza no Amazon ECR.

Usuário do serviço – se você usar o Amazon ECR para fazer sua tarefa, o administrador fornecerá as credenciais e as permissões de que você precisa. À medida que usar mais recursos do Amazon ECR para fazer seu trabalho, você poderá precisar de permissões adicionais. Entender como o acesso é gerenciado pode ajudar você a solicitar as permissões corretas ao seu administrador. Se não for possível acessar um recurso no Amazon ECR, consulte Resolução de problemas Amazon Elastic Container Registry Identidade e acesso (p. 73).

Administrador do serviço – se você for o responsável pelos recursos do Amazon ECR em sua empresa, você provavelmente terá acesso total ao Amazon ECR. Seu trabalho é determinar quais recursos do Amazon ECR seus funcionários devem acessar. Assim, é necessário enviar solicitações ao administrador do IAM para alterar as permissões dos usuários de seu serviço. Revise as informações nesta página para entender os conceitos básicos do IAM. Para saber mais sobre como sua empresa pode usar o IAM com o Amazon ECR, consulte Como Amazon Elastic Container Registry Trabalha com IAM (p. 64).

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 visualizar exemplos de políticas baseadas em identidade do Amazon ECR que podem ser usadas no IAM, consulte Amazon Elastic Container RegistryExemplos de políticas baseadas em identidade do (p. 69).

Autenticação com identidades

A autenticação é a forma como você faz login na AWS usando suas credenciais de identidade. Para obter mais informações sobre como fazer login usando o Console de gerenciamento da AWS, consulte A página de login e do console do IAM no Guia do usuário do IAM.

Você deve ser autenticado (fazer login na AWS) como o Usuário raiz da conta da AWS, um usuário do IAM, ou assumindo uma função do IAM. Também é possível usar a autenticação de logon único da sua empresa, ou até mesmo fazer login usando o Google ou o Facebook. Nesses casos, seu administrador configurou anteriormente a federação de identidades usando funções do IAM. Ao acessar a AWS usando credenciais de outra empresa, você estará assumindo uma função indiretamente.

Para fazer login diretamente no Console de gerenciamento da AWS, use sua senha com o e-mail do usuário raiz ou seu nome de usuário do IAM. É possível acessar a AWS de maneira programática usando seu usuário raiz ou as chaves de acesso do usuário do IAM. A AWS fornece ferramentas do SDK ou da linha de comando para assinar sua solicitação de forma criptográfica usando suas credenciais. Se você não utilizar ferramentas da AWS, cadastre a solicitação você mesmo. Faça isso usando o Signature versão 4, um protocolo para autenticação de solicitações de API de entrada. Para obter mais informações sobre a autenticação de solicitações, consulte Processo de cadastramento do Signature versão 4 na AWS General Reference.

Independentemente do método de autenticação usado, também pode ser exigido que você forneça informações adicionais de segurança. Por exemplo, a AWS recomenda o uso da autenticação multifator (MFA) para aumentar a segurança de sua conta. Para saber mais, consulte Usar a autenticação multifator (MFA) na AWS no Guia do usuário do IAM.

Usuário raiz da conta da AWS

Ao criar uma conta da AWS, você começa com uma única identidade de login que tenha acesso total a todos os recursos e serviços da AWS na conta. Essa identidade é chamada de AWS da conta da usuário

raiz e é acessada pelo login com o endereço de e-mail e a senha que você usou para criar a conta. Recomendamos que não use o usuário raiz para suas tarefas diárias, nem mesmo as administrativas. Em vez disso, siga as melhores práticas de uso do usuário raiz somente para criar seu primeiro usuário do IAM. Depois, armazene as credenciais usuário raiz com segurança e use-as para executar apenas algumas tarefas de gerenciamento de contas e de serviços.

IAMGrupos e usuários do

Um usuário do IAM é uma identidade em sua conta da AWS que tem permissões específicas para uma única pessoa ou um único aplicativo. Um usuário do IAM pode ter credenciais de longo prazo, como um nome de usuário e uma senha ou um conjunto de chaves de acesso. Para saber como gerar chaves de acesso, consulte Gerenciar chaves de acesso para usuários do IAM no Guia do usuário do IAM. Ao gerar chaves de acesso para um usuário do IAM, visualize e salve o par de chaves de maneira segura. Não será possível recuperar a chave de acesso secreta futuramente. Em vez disso, você deverá gerar outro par de chaves de acesso.

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 grupo chamado Administradores do IAM e atribuir a esse grupo permissões para administrar recursos do IAM.

Usuários são diferentes de funções. 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 as funções fornecem credenciais temporárias. Para saber mais, consulte Quando criar um usuário do IAM (em vez de uma função) no Guia do usuário do IAM.

IAMFunções do do

Uma função do IAM é uma identidade dentro de sua conta da AWS que tem permissões específicas. Ela é semelhante a um usuário do IAM, mas não está associada a uma pessoa específica. É possível assumir temporariamente uma função do IAM no Console de gerenciamento da AWS alternando funções. É possível assumir uma função chamando uma operação de API da AWS CLI ou da AWS, ou usando um URL personalizado. Para obter mais informações sobre os métodos para o uso de funções, consulte Usar funções do IAM no Guia do usuário do IAM.

As funções do IAM com credenciais temporária são úteis nas seguintes situações:

- Permissões temporárias para usuários do IAM – um usuário do IAM pode assumir uma função do IAM para obter temporariamente permissões diferentes para uma tarefa específica.
- Acesso de usuário federado – Em vez de criar um usuário do IAM, você pode usar identidades existentes do AWS Directory Service, do diretório de usuário da sua empresa ou de um provedor de identidades da Web. Estes são conhecidos como usuários federados. A AWS atribui uma função a um usuário federado quando o acesso é solicitado por meio de um provedor de identidades. Para obter mais informações sobre usuários federados, consulte Usuários federados e funções no Guia do usuário do IAM.
- Acesso entre contas – é possível usar uma função do IAM para permitir que alguém (um principal confiável) em outra conta acesse recursos em sua conta. As funções são a principal forma de conceder acesso entre contas. No entanto, alguns serviços da AWS permitem que você anexe uma política diretamente a um recurso (em vez de usar uma função como proxy). Para saber 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 a serviços da AWS – Uma função de serviço é uma função do IAM que um serviço assume para realizar ações em seu nome na sua conta. Ao configurar alguns ambientes de serviço da AWS, você deve definir uma função a ser assumida pelo serviço. Essa função de serviço deve incluir todas as permissões necessárias para o serviço acessar os recursos da AWS de que precisa. As funções de serviço variam de acordo com o serviço, mas muitas permitem que você escolha as permissões, desde

que atenda aos requisitos documentados para esse serviço. As funções de serviço fornecem acesso apenas dentro de sua conta e não podem ser usadas para conceder acesso a serviços em outras contas. Você pode criar, modificar e excluir uma função de serviço no IAM. Por exemplo, você pode criar uma função que permita que Amazon Redshift acesse um bucket do Amazon S3 em seu nome e carregue dados desse bucket em um cluster Amazon Redshift. Para obter mais informações, consulte Criar uma função para delegar permissões a um serviço da AWS no Guia do usuário do IAM.

- 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 que fazem solicitações de API da AWS CLI ou AWS. É preferível fazer isso do que armazenar chaves de acesso na instância do EC2. Para atribuir uma função da AWS a uma instância do EC2 e disponibilizá-la para todos os seus aplicativos, crie um perfil de instância que esteja anexado à instância. Um perfil de instância contém a função e permite que programas que estão em execução na instância do EC2 obtenham credenciais temporárias. Para mais informações, consulte Uso de uma função do IAM para conceder permissões aos aplicativos em execução nas instâncias do Amazon EC2 no Guia do usuário do IAM.

Para saber se você deve usar funções do IAM, consulte Quando criar uma função do IAM (em vez de um usuário) no Guia do usuário do IAM.

Gerenciamento do acesso usando políticas

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

Um administrador do IAM pode usar políticas para especificar quem tem acesso aos recursos da AWS e quais ações essas pessoas podem executar nesses recursos. Cada entidade do IAM (usuário ou função) começa sem permissões. Em outras palavras, por padrão, os usuários não podem fazer nada, nem mesmo alterar sua própria senha. Para dar permissão a um usuário para fazer algo, um administrador deve anexar uma política de permissões ao usuário. Ou o administrador pode adicionar o usuário a um grupo que tenha as permissões pretendidas. Quando um administrador concede permissões a um grupo, todos os usuários desse grupo recebem essas permissões.

As políticas do IAM definem permissões para uma ação, independentemente 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 funções do Console de gerenciamento da AWS, da AWS CLI ou da API da AWS.

Políticas com base em identidade

As políticas baseadas em identidade são documentos JSON de políticas de permissões que você pode anexar a uma entidade, como um usuário, função ou grupo do IAM. Essas políticas controlam quais ações cada identidade pode realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte Criar políticas do IAM no Guia do usuário do IAM.

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

Políticas baseadas em recursos

Políticas baseadas em recursos são documentos de política JSON que você anexa a um recurso, como um bucket do Amazon S3. Os administradores do serviço podem usar essas políticas para definir quais ações um principal especificado (função, usuário ou membro da conta) pode executar nesse recurso e sob quais condições. As políticas baseadas em recurso são políticas em linha. Não há políticas baseadas em recurso gerenciadas.

Outros tipos de política

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

- Limites de permissões – um limite de permissões é um 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 função do IAM). É possível definir um limite de permissões para uma entidade. As permissões resultantes são a interseção das políticas baseadas em identidade da entidade e seus limites de permissões. As políticas baseadas em recurso que especificam o usuário ou a função 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 – Service control policies) – SCPs são políticas JSON que especificam o máximo de permissões para uma organização ou unidade organizacional (UO) no AWS Organizations. O AWS Organizations é um serviço para agrupamento e gerenciamento central das várias contas da AWS que sua empresa possui. Se você habilitar todos os recursos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as contas. O SCP limita as permissões para entidades em contas-membro, incluindo cada Usuário raiz da conta da AWS. Para obter mais informações sobre Organizações e SCPs, consulte Como SCPs funcionam no Guia do usuário do AWS Organizations.
- Políticas de sessão – as 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 da função e das políticas de sessão. As permissões também podem ser provenientes de uma política baseada em recurso. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações, consulte Políticas de sessão no Guia do usuário do IAM.

Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como a AWS determina se deve permitir uma solicitação quando vários tipos de política estão envolvidos, consulte Lógica de avaliação de políticas no Guia do usuário do IAM.

Como Amazon Elastic Container Registry Trabalha com IAM

Antes de utilizar IAM para gerir o acesso a Amazon ECR, deve compreender o que IAM estão disponíveis para utilização com Amazon ECR. Para obter uma visão de alto nível de como Amazon ECR e outros AWS serviços trabalham com IAM, consulte AWS Serviços que trabalham com IAM no Guia do usuário do IAM.

Tópicos

- Amazon ECR IAM Funções (p. 67)

Amazon ECRPolíticas baseadas em identidade do

Com IAM políticas baseadas na identidade, pode especificar acções e recursos permitidos ou negados, bem como as condições sob as quais as acções são permitidas ou negadas. Amazon ECR suporta acções específicas, recursos e teclas de condição. Para saber mais sobre todos os elementos que utiliza numa política JSON, consulte IAM Referência dos elementos da política JSON no Guia do usuário do IAM.

Actions

O elemento `Action` de uma política baseada em identidade do IAM descreve a ação ou ações específicas que serão permitidas ou negadas pela política. As ações de política geralmente têm o mesmo nome que a operação de API da AWS associada. A ação é usada em uma política para conceder permissões para executar a operação associada.

As ações de políticas no Amazon ECR usam o seguinte prefixo antes da ação: . `ecr:`. Por exemplo, para conceder a alguém permissão para criar um Amazon ECR repositório com o Amazon ECR `CreateRepository` Operação API, inclui o `ecr:CreateRepository` na sua apólice. As declarações da política devem incluir um `Action` ou `NotAction` elemento. Amazon ECR define o seu próprio conjunto de ações que descrevem tarefas que pode executar com este serviço.

Para especificar várias ações em uma única declaração, separe-as com vírgulas, conforme o seguinte.

```
"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 ação a seguir.

```
"Action": "ecr:Describe*"
```

Para ver uma lista de Amazon ECR ações, ver Teclas de ações, recursos e condições para Amazon Elastic Container Registry no Guia do usuário do IAM.

Resources

O elemento `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou um elemento `NotResource`. Você especifica um recurso usando um ARN ou usando o caractere curinga (*) para indicar que a instrução se aplica a todos os recursos.

Um recurso do repositório do Amazon ECR tem o seguinte ARN:

```
arn:${Partition}:ecr:${Region}:${Account}:repository/${Repository-name}
```

Para mais informações sobre o formato de arcos, consulte Nomes de Recursos Amazon (arcos) e AWS Nomes de serviço.

Por exemplo, para especificar o `my-repo` repositório no `us-east-1` Região na sua declaração, utilize 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 de Amazon ECR tipos de recursos e seus arcos, consulte Recursos definidos por Amazon Elastic Container Registry no Guia do usuário do IAM. Para aprender com que ações pode especificar o ARN de cada recurso, consulte Ações definidas por Amazon Elastic Container Registry.

Chaves de condição

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

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

Você também pode usar variáveis de espaço reservado ao especificar as condições. Por exemplo, você pode conceder a um usuário do IAM permissão para acessar um recurso 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.

Amazon ECRA define seu próprio conjunto de chaves de condição e também oferece suporte ao uso de algumas chaves de condição globais. Para ver todos AWS teclas de condição global, consulte AWS Teclas de contexto da condição global no Guia do usuário do IAM.

Mais Amazon ECR as ações apoiam o `aws:ResourceTag` e `ecr:ResourceTag` teclas de condição. Para obter mais informações, consulte Usar controle de acesso baseado em tags (p. 72).

Para ver uma lista de Amazon ECR teclas de condição, consulte Teclas de condição definidas por Amazon Elastic Container Registry no Guia do usuário do IAM. Para aprender com quais as acções e recursos que pode utilizar, consulte Ações definidas por Amazon Elastic Container Registry.

Examples

Para ver exemplos de políticas baseadas em identidade do Amazon ECR, consulte Amazon Elastic Container RegistryExemplos de políticas baseadas em identidade do (p. 69).

Amazon ECRPolíticas baseadas em recurso do

As políticas baseadas em recursos são documentos da política JSON que especificam as acções que um responsável especificado pode realizar num Amazon ECR recursos e em que condições. Amazon ECR apoia políticas de permissões baseadas em recursos para Amazon ECR repositórios. As políticas baseadas em recursos permitem conceder permissão de uso a outras contas por recurso. Também pode utilizar uma política baseada em recursos para permitir uma AWS para aceder ao seu Amazon ECR repositórios.

Para ativar o acesso à conta cruzada, pode especificar uma conta inteira ou IAM entidades noutra conta como principal numa política baseada em recursos. Adicionar um principal entre contas à política baseada em recurso é apenas metade da tarefa de estabelecimento da relação de confiança. Quando o principal e o recurso estão em diferentes contas da AWS, você também deve conceder à entidade principal

permissão 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 um principal na mesma conta, nenhuma política baseada em identidade adicional será necessária. Para mais informações, consulte Como IAM As funções diferem das políticas baseadas em recursos no Guia do usuário do IAM.

O Amazon ECR suporte suporta apenas um tipo de política baseada em recursos chamada de política de repositório, que está ligado 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 (p. 20).

Examples

Para visualizar exemplos de políticas baseadas em recursos do Amazon ECR, consulte .Exemplos de política de repositório (p. 23),

Autorização baseada em Amazon ECR Etiquetas

Você pode anexar tags a recursos do Amazon ECR ou passar tags em uma solicitação ao Amazon ECR. Para controlar o acesso com base nas etiquetas, fornece informações de etiqueta no elemento de condição de uma política utilizando o `ecr:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`*, ou `aws:TagKeys` teclas de condição. Para obter mais informações sobre recursos de marcação do Amazon ECR, consulte .Marcar um Amazon ECR repositório (p. 26).

Para visualizar um exemplo de política baseada em identidade que visa limitar o acesso a um recurso baseado nas tags desse recurso, consulte Usar controle de acesso baseado em tags (p. 72).

Amazon ECR IAM Funções

Um IAM função é uma entidade no seu AWS conta com permissões específicas.

Usar credenciais temporárias com o Amazon ECR

Você pode usar credenciais temporárias para fazer login com federação, assumir uma função do IAM ou assumir uma função entre contas. Obtém credenciais de segurança temporárias ligando AWS STS Operações API como permerole ou getfederationtoken.

Amazon ECRO oferece suporte ao uso de credenciais temporárias.

Funções vinculadas ao serviço

Funções ligadas ao serviço permitir AWS serviços para aceder a recursos noutros serviços para concluir uma acção em seu nome. As funções vinculadas ao serviço aparecem em sua conta do IAM e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não pode editar as permissões para funções vinculadas ao serviço.

Amazon ECRO não oferece suporte às funções vinculadas ao serviço.

Políticas gerenciadas do Amazon ECR

O Amazon ECR fornece várias políticas gerenciadas que você pode anexar aos usuários do IAM ou às instâncias do EC2 que permitem níveis de controle diferentes nas operações de API e nos recursos do Amazon ECR. Você pode aplicar essas políticas diretamente ou usá-las como ponto de partida para criar suas próprias políticas. Para obter mais informações sobre cada uma das operações de API mencionadas nessas políticas, consulte Ações no Amazon Elastic Container Registry API Reference.

Tópicos

- AmazonEC2ContainerRegistryPowerUser (p. 68)
- AmazonEC2ContainerRegistryReadOnly (p. 68)

AmazonEC2ContainerRegistryFullAccess

Esta política gerida é um ponto de partida para os clientes que pretendem fornecer a um usuário ou a uma função do IAM acesso total de administrador para gerenciar seu 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 eventos do CloudTrail, e o Amazon ECR é integrados ao AWS CloudTrail para exibir os eventos da política de ciclo de vida de um cliente diretamente no console do Amazon ECR. A política gerenciada `AmazonEC2ContainerRegistryFullAccess` do IAM inclui a permissão `cloudtrail:LookupEvents` para facilitar esse comportamento.

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

AmazonEC2ContainerRegistryPowerUser

Essa política gerenciada permite o acesso de usuário avançado ao Amazon ECR, o que oferece acesso de leitura e gravação nos repositórios, mas não permite que os usuários excluam repositórios ou alterem os documentos de política aplicados a eles.

```
{
 "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:InitiateLayerUpload",
   "ecr:UploadLayerPart",
   "ecr:CompleteLayerUpload",
   "ecr:PutImage"
  ],
  "Resource": "*"
 }]
}
```

AmazonEC2ContainerRegistryReadOnly

Essa política gerenciada permite o acesso somente leitura ao Amazon ECR, como a capacidade de listar repositórios e imagens nos repositórios, além de extrair imagens do Amazon ECR com a CLI do Docker.

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

Amazon Elastic Container RegistryExemplos de políticas baseadas em identidade do

Por predefinição, IAM utilizadores e funções não têm permissão para criar ou modificar Amazon ECR recursos. Também não conseguem realizar tarefas utilizando o Console de gerenciamento da AWS, AWS CLI, ou AWS API. Um IAM o administrador deve criar IAM políticas que concedem aos utilizadores e funções permissão para realizar operações específicas da API sobre os recursos específicos de que necessitam. O administrador deve anexar essas políticas aos usuários ou grupos do IAM que exigem essas permissões.

Para aprender a criar um IAM política baseada na identidade utilizando estes exemplos de documentos da política JSON, consulte Criar políticas no separador JSON no Guia do usuário do IAM.

Tópicos

Melhores práticas de políticas

As políticas baseadas em identidade são muito eficientes. Elas determinam se alguém pode criar, acessar ou excluir recursos do Amazon ECR em sua conta. Essas ações podem incorrer em custos para sua conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece usando políticas gerenciadas pela AWS – para começar a usar o Amazon ECR rapidamente, use as políticas gerenciadas pela AWS para conceder a seus funcionários as permissões de que precisam. Essas políticas já estão disponíveis em sua conta e são mantidas e atualizadas pela AWS. Para obter mais informações, consulte Conceitos básicos do uso de permissões com políticas gerenciadas pela AWS no Guia do usuário do IAM.
- Conceder privilégio mínimo – ao criar políticas personalizadas, conceda apenas as permissões necessárias para executar uma tarefa. Comece com um conjunto mínimo de permissões e conceda permissões adicionais conforme necessário. Fazer isso é mais seguro do que começar com permissões que são muito lenientes e tentar restringi-las posteriormente. Para obter mais informações, consulte Conceder privilégio mínimo, no Guia do usuário do IAM.
- Habilitar o MFA para operações confidenciais – para segurança adicional, exija que os usuários do IAM usem a autenticação multifator (MFA) para acessar recursos ou operações de API confidenciais. Para

obter mais informações, consulte Usar a autenticação multifator (MFA) na AWS no Guia do usuário do IAM.

- Usar condições de política para segurança adicional – na medida do possível, defina as condições sob as quais suas políticas baseadas em identidade permitem o acesso a um recurso. Por exemplo, você pode gravar condições para especificar um intervalo de endereços IP permitidos do qual a solicitação deve partir. Você também pode escrever condições para permitir somente solicitações em uma data especificada ou período ou para exigir o uso de SSL ou MFA. Para obter mais informações, consulte Elementos da política JSON do IAM: condição no Guia do usuário do IAM.

Usar o Amazon ECR Console

Para acessar o console do Amazon Elastic Container Registry, é necessário ter um conjunto mínimo de permissões. Estas permissões devem permitir-lhe listar e ver detalhes sobre o Amazon ECR recursos no seu AWS conta. Se você criar uma política baseada em identidade que seja mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para entidades (usuários ou funções do IAM) com essa política.

Para garantir que essas entidades ainda podem utilizar o Amazon ECR consola, adicionar `AmazonEC2ContainerRegistryReadOnly` AWS política gerida às entidades. Para mais informações, consulte Adicionar permissões a um utilizador no Guia 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"
  ],
  "Resource": "*"
 }]
}
```

Não precisa de permitir permissões mínimas da consola para utilizadores que estão a efectuar chamadas apenas para o AWS CLI ou o 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 você pode criar uma política que permite que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou de forma programática usando a AWS CLI ou a API da 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": "*"
        }
    ]
}
```

Aceder a um Amazon ECR Repositório

Neste exemplo, pretende conceder um IAM utilizador na sua AWS acesso a uma das suas Amazon ECR repositórios, `my-repo`. Também pretende permitir que o utilizador prima, puxe 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 de API CreateRepository do Amazon ECR permite especificar tags ao criar o repositório. Para obter mais informações, consulte Marcar um Amazon ECR repositório (p. 26).

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 do IAM 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 do IAM 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"
                }
            }
        }
    ]
}
```

Resolução de problemas Amazon Elastic Container Registry Identidade e acesso

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

Tópicos

Não tenho autorização para executar uma ação no Amazon ECR

Se o Console de gerenciamento da AWS informar que você não está autorizado a executar uma ação, você deverá entrar em contato com o administrador para obter assistência. O administrador é a pessoa que forneceu a você o seu nome de usuário e senha.

O seguinte erro ocorre quando o `mateojackson` IAM o utilizador tenta utilizar a consola para ver detalhes sobre um repositório mas não tem `ecr:DescribeRepositories` permissões.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 ecr:DescribeRepositories on resource: my-repo
```

Neste caso, Mateo pede ao seu administrador para atualizar as suas políticas para permitir que aceda ao `my-repo` recurso utilizando o `ecr:DescribeRepositories` ação.

Não estou autorizado a executar iam:PassRole

Se você receber uma mensagem de erro informando que você não está autorizado a executar a ação `iam:PassRole`, entre em contato com o administrador para obter assistência. O administrador é a pessoa que forneceu a você o seu nome de usuário e senha. Peça a essa pessoa para atualizar suas políticas para permitir que você passe uma função para o Amazon ECR.

Alguns serviços da AWS permitem que você passe uma função existente para o serviço, em vez de criar uma nova função de serviço ou função vinculada ao serviço. Para fazer isso, um usuário deve ter permissões para passar a função para o serviço.

O erro de exemplo a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta usar o console para executar uma ação no Amazon ECR. No entanto, a ação exige que o serviço tenha permissões concedidas por uma função de serviço. Mary não tem permissões para passar a função para o serviço.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform: iam:PassRole
```

Neste caso, Mary pede ao administrador para atualizar suas políticas para permitir que ela execute a ação `iam:PassRole`.

Quero visualizar minhas chaves de acesso

Depois de criar suas chaves de acesso de usuário do IAM, é possível visualizar seu ID de chave de acesso a qualquer momento. No entanto, você não pode visualizar sua chave de acesso secreta novamente. Se você perder sua chave secreta, crie um novo par de chaves de acesso.

As chaves de acesso consistem em duas partes: um ID de chave de acesso (por exemplo, `AKIAIOSFODNN7EXAMPLE`) e uma chave de acesso secreta (por exemplo, `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`). Como um nome de usuário e uma senha, você deve usar o ID da chave de acesso e a chave de acesso secreta em conjunto para autenticar suas solicitações. Gerencie suas chaves de acesso de forma tão segura quanto você gerencia seu nome de usuário e sua senha.

> **Important**
>
> Não forneça as chaves de acesso a terceiros, mesmo que seja para ajudar a encontrar seu ID de usuário canônico. Ao fazer isso, você pode dar a alguém acesso permanente à sua conta.

Ao criar um par de chaves de acesso, você é solicitado a guardar o ID da chave de acesso e a chave de acesso secreta em um local seguro. A chave de acesso secreta só está disponível no momento em que é criada. Se você perder sua chave de acesso secreta, você deverá adicionar novas chaves de acesso para seu usuário do IAM. Você pode ter no máximo duas chaves de acesso. Se você já tiver duas, você deverá excluir um par de chaves para poder criar um novo. Para visualizar as instruções, consulte Gerenciar chaves de acesso no Guia do usuário do IAM.

Sou administrador e desejo conceder acesso ao para outros usuários.Amazon ECR

Para permitir que outros usuários acessem o Amazon ECR, é necessário criar uma entidade do IAM (usuário ou função) para a pessoa ou o aplicativo que precisa do acesso. Eles usarão as credenciais dessa entidade para acessar a AWS. Você deve anexar uma política à entidade que concede a eles as permissões corretas no Amazon ECR.

Para começar a usar imediatamente, consulte Criar os primeiros usuário e grupo delegados do IAM no Guia do usuário do IAM.

Quero permitir pessoas fora do meu AWS Conta para aceder ao meu Amazon ECR Recursos

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

Para saber mais, consulte o seguinte:

- Para saber se o Amazon ECR oferece suporte a esses recursos, consulte Como Amazon Elastic Container Registry Trabalha com IAM (p. 64).

- Para saber como conceder acesso aos seus recursos em todas as contas da AWS pertencentes a você, consulte Conceder acesso a um usuário do IAM em outra conta da AWS pertencente a você no Guia do usuário do IAM.
- Para saber como conceder acesso aos seus recursos para contas da AWS de terceiros, consulte Conceder acesso a contas da AWS pertencentes a terceiros no Guia do usuário do IAM.
- Para saber como fornecer acesso por meio de federação de identidades, consulte Fornecer 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 recurso para acesso entre contas, consulte Como as funções do IAM diferem de políticas baseadas em recursos no Guia do usuário do IAM.

Proteção de dados no Amazon ECR

Amazon Elastic Container Registry (Amazon ECR) está em conformidade com o AWS modelo de responsabilidade partilhada, que inclui regulamentos e diretrizes para proteção de dados. AWS é responsável por proteger a infraestrutura global que executa todos os AWS serviços. AWS mantém o controlo sobre os dados alojados nesta infraestrutura, incluindo os controlos de configuração de segurança para lidar com o conteúdo do cliente e os dados pessoais. AWS clientes e parceiros APN, agindo como responsáveis pelo tratamento de dados ou subcontratantes, são responsáveis por quaisquer dados pessoais que os mesmos coloquem no AWS Nuvem.

Para fins de proteção de dados, recomendamos que proteja AWS credenciais de conta e configurar contas de utilizador individuais com AWS Identity and Access Management (IAM), para que cada utilizador receba apenas as permissões necessárias para cumprir os seus deveres profissionais. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA) com cada conta.
- Utilizar SSL/TLS para comunicar com AWS recursos.
- Configure o registro em log das atividades da API e do usuário com o AWS CloudTrail.
- Utilização AWS soluções de encriptação, juntamente com todos os controlos de segurança predefinidos no AWS serviços.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajudam a localizar e proteger dados pessoais que são armazenados no ()Amazon S3.

Recomendamos vivamente que nunca coloque informações de identificação sensíveis, tais como os números de conta dos seus clientes, em campos de forma livre, tais como Nome campo. Isto inclui quando trabalha com Amazon ECR ou outro AWS serviços utilizando a consola, API, AWS CLI, ou AWS ks. Todos os dados inseridos no Amazon ECR ou em outros serviços poderão ser selecionados para inclusão em logs de diagnóstico. Ao fornecer um URL para um servidor externo, não inclua informações de credenciais no URL para validar a solicitação a esse servidor.

Para mais informações sobre proteção de dados, consulte o AWS Modelo de Responsabilidade Partilhada e RGPD publicação do blogue no AWS Blogue de segurança.

Tópicos

Criptografia em repouso

Amazon ECR grava imagens em Amazon S3 baldes que Amazon ECR gere. Por predefinição, Amazon ECR utiliza encriptação lado do servidor com Amazon S3-chaves de encriptação geridas que encriptam os seus dados em repouso utilizando um algoritmo de encriptação AES-256. Isto não requer qualquer ação

da sua parte e é oferecido sem custos adicionais. Para mais informações, consulte Proteção de dados utilizando encriptação do lado do servidor com chaves de encriptação geridas pela Amazon S3 (SSE-S3) no Guia do desenvolvedor do Amazon Simple Storage Service.

Para mais controlo sobre a encriptação para o seu Amazon ECR repositórios, pode utilizar encriptação do lado do servidor com chaves mestre do cliente (cmks) armazenadas em AWS Key Management Service (AWS KMS). Quando utiliza AWS KMS para encriptar os seus dados, pode utilizar a predefinição AWS-gerido pelo CMK, que é gerido por Amazon ECR, ou especifique o seu próprio CMK (referido como um CMK gerido pelo cliente). Para mais informações, consulte Proteção de dados utilizando encriptação do lado do servidor com cmks armazenados no serviço de gestão chave AWS (SSE-KMS) no Guia do desenvolvedor do Amazon Simple Storage Service.

Cada Amazon ECR o repositório tem uma configuração de encriptação, que é definida quando o repositório é criado. Pode utilizar diferentes configurações de encriptação em cada repositório. Para obter mais informações, consulte Criar um repositório (p. 17).

Quando um repositório é criado com AWS KMS codificação ativada, é utilizado um CMK para encriptar o conteúdo do repositório. Além disso, Amazon ECR adiciona um AWS KMS conceder ao CMK Amazon ECR como o capital do subvencionado.

O que se segue proporciona uma compreensão de alto nível sobre como Amazon ECR está integrado com AWS KMS para encriptar e desencriptar os seus repositórios:

1. Ao criar um repositório, Amazon ECR envia um descrição chamada para AWS KMS para validar e recuperar o Nome do Recurso Amazon (ARN) do CMK especificado na configuração da encriptação.
2. Amazon ECR envia dois concessão de brindes pedidos para AWS KMS para criar subsídios no CMK para permitir Amazon ECR para encriptar e desencriptar dados utilizando a tecla de dados.
3. Ao carregar uma imagem, chave geratedata pedido é feito para AWS KMS que especifica o CMK a utilizar para encriptar a camada de imagens e o manifesto.
4. AWS KMS gera uma nova chave de dados, encripta-a sob o CMK especificado e envia a chave de dados encriptada para ser armazenada com os metadados da camada de imagens e o manifesto da imagem.
5. Ao puxar uma imagem, Descriptar pedido é feito para AWS KMS, especificando a chave de dados encriptada.
6. AWS KMSO descriptografa a chave de dados criptografada e envia a chave de dados descriptografada ao Amazon S3.
7. A chave de dados utilizada para desencriptar a camada de imagens antes de a camada de imagem ser puxada.
8. Quando um repositório é eliminado, Amazon ECR envia dois reregante pedidos para AWS KMS para retirar os subsídios criados para o repositório.

Considerations

Os pontos seguintes devem ser considerados ao utilizar AWS KMS encriptação com Amazon ECR.

- Se criar o seu Amazon ECR repositório com encriptação KMS e não especificar um CMK, Amazon ECR utiliza um AWS-CMK gerido com o alias `aws/ecr` por predefinição. Este CMK é criado na sua conta pela primeira vez que cria um repositório com encriptação KMS ativada.
- Quando utiliza encriptação KMS com o seu próprio CMK, a chave tem de existir na mesma região que o seu repositório.
- AWS KMS obedece um limite de 500 subsídios por CMK. Como resultado, existe um limite de 500 Amazon ECR repositórios que podem ser encriptados por CMK.
- As subvenções que Amazon ECR cria em seu nome não deve ser revogado. Se revogar o subsídio que dá Amazon ECR permissão para utilizar o AWS KMS na sua conta, Amazon ECR não é possível aceder a estes dados, encriptar novas imagens para o repositório ou descriá-las quando forem retiradas.

Quando revoga uma subvenção para Amazon ECR, a alteração ocorre imediatamente. Para revogar direitos de acesso, deve eliminar o repositório em vez de revogar o subsídio. Quando um repositório é eliminado, Amazon ECR recolhe as subvenções em seu nome.

- Existe um custo associado à utilização AWS KMS teclas. Para mais informações, consulte AWS Key Management Service preços.

Obrigatório IAM permissões

Ao criar ou eliminar um Amazon ECR repositório com encriptação do lado do servidor utilizando AWS KMS, as permissões necessárias dependem da chave mestre de cliente (CMK) que está a utilizar.

Obrigatório IAM permissões ao utilizar o AWS gerido CMK para Amazon ECR

Por predefinição, quando AWS KMS a encriptação está ativada para um Amazon ECR mas não é especificado nenhum CMK, o AWS-controlo CMK para Amazon ECR é utilizado. Quando o AWS-controlo CMK para Amazon ECR é utilizado para encriptar um repositório, qualquer principal que tenha permissão para criar um repositório também pode permitir AWS KMS encriptação no repositório. No entanto, o IAM principal que apaga o repositório deve ter o `kms:RetireGrant` permissão. Isto permite a retirada dos subsídios que foram adicionados ao AWS KMS chave quando o repositório foi criado.

O exemplo seguinte IAM política pode ser adicionada como uma política em linha a um utilizador para garantir que tem as permissões mínimas necessárias para eliminar um repositório com encriptação ativada. O AWS KMS chave utilizada para encriptar o repositório pode ser especificada utilizando o parâmetro de recurso.

```
{
    "Version": "2012-10-17",
    "Id": "ecr-kms-permissions",
    "Statement": [
        {
            "Sid": "Allow access to retire the grants associated with the key",
            "Effect": "Allow",
            "Action": [
                "kms:RetireGrant"
            ],
            "Resource": "arn:aws:kms:us-
west-2:111122223333:key/b8d9ae76-080c-4043-92EXAMPLE"
        }
    ]
}
```

Obrigatório IAM quando se utiliza um CMK gerido pelo cliente

Ao criar um repositório com AWS KMS encriptação ativada utilizando um CMK gerido pelo cliente, existem permissões necessárias para a política-chave do CMK e para o IAM política para o utilizador ou função que cria o repositório.

Ao criar o seu próprio CMK, pode utilizar a política chave predefinida AWS KMS cria ou pode especificar o seu próprio. Para garantir que o CMK gerido pelo cliente continua a ser gerido pelo titular da conta, a política-chave para o CMK deve permitir todos AWS KMS ações para o utilizador raiz da conta. Podem ser adicionadas autorizações adicionais com direitos de autor à política chave, mas no mínimo, o utilizador raiz deve receber permissões para gerir o CMK. Permitir que o CMK seja utilizado apenas para pedidos que originem Amazon ECR, pode utilizar o kms:viaservice-chave com o `ecr.<region>.amazonaws.com` valor.

A seguinte política chave de exemplo dá ao AWS conta (utilizador raiz) que detém o acesso total CMK ao CMK. Para mais informações sobre esta política chave de exemplo, consulte Permite o acesso à conta AWS e permite as políticas de IAM no AWS Key Management Service Developer Guide.

```
{
    "Version": "2012-10-17",
    "Id": "ecr-key-policy",
    "Statement": [
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        }
    ]
}
```

O IAM utilizador, IAM função ou AWS a conta que cria os seus repositórios deve ter o `kms:CreateGrant`, `kms:RetireGrant`, e `kms:DescribeKey` para além do necessário Amazon ECR permissões.

> **Note**
>
> O `kms:RetireGrant` a autorização tem de ser adicionada ao IAM política do utilizador ou função que cria o repositório. O `kms:CreateGrant` e `kms:DescribeKey` podem ser adicionadas permissões à política-chave do CMK ou do IAM política de utilizador ou função que cria o repositório. Para mais informações sobre como AWS KMS tarefas de trabalho, ver Permissões de API AWS KMS: Referência de ações e recursos no AWS Key Management Service Developer Guide.

O exemplo seguinte IAM a política pode ser adicionada como uma política em linha a um utilizador para garantir que tem as permissões mínimas necessárias para criar um repositório com encriptação ativada e eliminar o repositório quando terminar com ele. O AWS KMS chave utilizada para encriptar o repositório pode ser especificada utilizando o parâmetro de recurso.

```
{
    "Version": "2012-10-17",
    "Id": "ecr-kms-permissions",
    "Statement": [
        {
            "Sid": "Allow access to create and retire the grants associated with the key as
 well as describe the key",
            "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 utilizador liste os ficheiros na consola ao criar um repositório

Quando utilizar o Amazon ECR a consola para criar um repositório, pode conceder permissões para permitir que um utilizador liste as colunas geridas pelo cliente na região ao permitir a encriptação para o repositório. O seguinte IAM exemplo de política mostra as permissões necessárias para listar os seus cmks e alias quando utilizar a consola.

```
{
```

```
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "kms:ListKeys",
      "kms:ListAliases",
      "kms:DescribeKey"
    ],
    "Resource": "*"
  }
}
```

Monitorização Amazon ECR interação com AWS KMS

Pode usar AWS CloudTrail para acompanhar os pedidos que Amazon ECR envia para AWS KMS em seu nome. As entradas de registo no CloudTrail o registo contém uma tecla de contexto de encriptação para torná-las mais facilmente identificáveis.

Amazon ECRContexto de criptografia do

Um contexto de encriptação é um conjunto de pares de valores chave que contêm dados não secretos arbitrários. Quando você inclui um contexto de criptografia em uma solicitação para criptografar dados, o AWS KMS vincula de forma criptográfica o contexto de criptografia aos dados criptografados. Para descriptografar os dados, você deve passar o mesmo contexto de criptografia.

No seu chave geratedata e Descriptar pedidos para AWS KMS, Amazon ECR utiliza um contexto de encriptação com dois nomes–pares de valores que identificam o repositório e Amazon S3 medidor a ser utilizado. Isso é mostrado no exemplo a seguir. Os nomes não variam, mas os valores de contexto de encriptação 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"
}
```

Pode utilizar o contexto de encriptação para identificar estas operações criptográficas em registos e registos de auditoria, tais como AWS CloudTrail e Amazon CloudWatch Logs, e como condição para autorização em políticas e subvenções.

O contexto de criptografia do Amazon ECR consiste em dois pares de nome e valor.

- aws:s3:arn – O primeiro nome–o par de valores identifica o balde. A chave é `aws:s3:arn`. O valor é o Nome do Recurso Amazon (ARN) do Amazon S3 balde.

  ```
  "aws:s3:arn": "ARN of an Amazon S3 bucket"
  ```

  Por exemplo, se o ARN do balde for `arn:aws:s3:::us-west-2-starport-manifest-bucket/EXAMPLE1-90ab-cdef-fedc-ba987BUCKET1/sha256:a7766145a775d39e53a713c75b6fd6d318740e70327aaa3ed5d09e0ef33fc3df`, o contexto de encriptação incluirá o seguinte par.

  ```
  "arn:aws:s3:::us-west-2-starport-manifest-bucket/EXAMPLE1-90ab-cdef-fedc-ba987BUCKET1/sha256:a7766145a775d39e53a713c75b6fd6d318740e70327aaa3ed5d09e0ef33fc3df"
  ```

- aws:ecr:arn – O segundo nome–o par de valores identifica o Nome do Recurso 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 for `arn:aws:ecr:us-west-2:111122223333:repository/repository-name`, o contexto de encriptação incluirá o seguinte par.

```
"aws:ecr:arn": "arn:aws:ecr:us-west-2:111122223333:repository/repository-name"
```

Troubleshooting

Ao eliminar um Amazon ECR repositório com a consola, se o repositório for eliminado com sucesso, mas Amazon ECR não é possível retirar os subsídios adicionados ao seu CMK para o seu repositório, 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 isto ocorrer, pode reformar-se AWS KMS para o repositório próprio.

Para se reformar AWS KMS subsídios para um repositório manualmente

1. Listar as subvenções para o AWS KMS chave utilizada para o repositório. O `key-id` o valor está incluído no erro que recebe da consola. Também pode utilizar o `list-keys` comando para listar as colunas geridas pela AWS e as operações geridas pelo cliente numa região específica na 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 de Recursos Amazon (ARN) do seu repositório. Isto pode ser utilizado para determinar qual o subsídio adicionado à chave que pretende reformar. O `GrantId` o valor será utilizado ao retirar o subsídio no próximo passo.
2. Retirar cada subsídio para o AWS KMS chave adicionada para o repositório. Substituir o valor para *`GrantId`* com a ID do subsídio da saída do passo anterior.

   ```
   aws kms retire-grant \
        --key-id b8d9ae76-080c-4043-9237-c815bfc21dfc \
        --grant-id GrantId \
        --region us-west-2
   ```

Validação de conformidade do Amazon Elastic Container Registry

Auditores independentes avaliam a segurança e a conformidade do Amazon Elastic Container Registry como parte de vários programas de conformidade da AWS. Isso inclui SOC, PCI, HIPAA e outros.

Para obter uma lista de produtos da AWS no escopo de programas de conformidade específicos, consulte Produtos da AWS no escopo por programa de conformidade. Para obter informações gerais, consulte Programas de conformidade da AWS.

Você pode fazer download de relatórios de auditoria de terceiros usando o AWS Artifact. Para obter mais informações, consulte Fazer download de relatórios no AWS Artifact.

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

- Guias Quick Start de segurança e conformidade – esses guias de implantação abordam as considerações de arquitetura e fornecem etapas para implantação de ambientes de linha de base focados em conformidade e segurança na AWS.
- Whitepaper Arquitetura para segurança e conformidade com HIPAA – esse whitepaper descreve como as empresas podem usar a AWS para criar aplicativos em conformidade com a HIPAA.
- Recursos de conformidade da AWS – esta coleção de manuais e guias pode ser aplicada ao seu setor e local.
- Avaliar recursos com regras no Guia do desenvolvedor do AWS Config – o serviço do AWS Config avalia como suas configurações de recursos estão em conformidade com práticas internas, diretrizes do setor e regulamentos.
- AWS Security Hub – esse serviço da AWS fornece uma visão abrangente do estado da segurança na AWS que ajuda você a verificar sua conformidade com padrões e melhores práticas de segurança do setor.

Segurança da infraestrutura no Amazon Elastic Container Registry

Como um serviço gerenciado, o Amazon Elastic Container Registry é protegido pelos procedimentos de segurança da rede global da AWS que estão descritos no whitepaper Amazon Web Services: visão geral dos processos de segurança.

Use chamadas de API publicadas pela AWS para acessar o Amazon ECR por meio da rede. Os clientes devem oferecer suporte a Transport Layer Security (TLS) 1.0 ou posterior. Recomendamos TLS 1.2 ou posterior. Os clientes também devem ter suporte a pacotes de criptografia com sigilo de encaminhamento perfeito (PFS) como Ephemeral Diffie-Hellman (DHE) ou Ephemeral Elliptic Curve Diffie-Hellman (ECDHE). A maioria dos sistemas modernos como Java 7 e versões posteriores oferece suporte a esses modos.

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

É possível chamar essas operações de API de qualquer local da rede, mas o Amazon ECR não oferece suporte a políticas de acesso baseadas em recurso, que podem incluir restrições com base no endereço IP de origem. Também é possível usar políticas específicas do Amazon ECR para controlar o acesso de Amazon Virtual Private Cloud (Amazon VPC) endpoints ou de VPCs específicas. Efetivamente, isso isola o acesso à rede para um determinado recurso do Amazon ECR somente da VPC específica dentro da rede da AWS. Para obter mais informações, consulte Amazon ECR configurar os parâmetros de avaliação VPC (AWS privaçãoligação) (p. 81).

Amazon ECR configurar os parâmetros de avaliação VPC (AWS privaçãoligação)

Você pode melhorar a postura de segurança da sua VPC configurando o Amazon ECR para usar um VPC endpoint de interface. Os parâmetros de VPC são alimentados por AWS privatelink, uma tecnologia que lhe permite aceder de forma privada Amazon ECR através de endereços IP privados. AWS O PrivateLink

limita todo o tráfego de rede entre sua VPC e o Amazon ECR para a rede da Amazon. Você não precisa de um gateway da Internet, de um dispositivo NAT ou de um gateway privado virtual.

Para mais informações sobre AWS privatelink e endpoints VPC, consulte Parâmetros VPC no Guia do usuário da Amazon VPC.

Considerações para Amazon ECR Parâmetros VPC

Antes de configurar VPC endpoints para o Amazon ECR, fique atento às seguintes considerações:

- Para permitir que o seu Amazon ECS tarefas que utilizam o EC2 tipo de lançamento para extrair imagens privadas de Amazon ECR, certifique-se de que também cria os terminais VPC da interface para Amazon ECS. Para mais informações, consulte Configurar os parâmetros de avaliação VPC (AWS privaçãoligação) no Amazon Elastic Container Service Developer Guide.

  Important

  Amazon ECS tarefas que utilizam o Fargate tipo de lançamento não requer o Amazon ECS interface de parâmetros de avaliação de VPC.
- Amazon ECS tarefas utilizando o Fargate tipo de lançamento e plataforma, versão 1.3.0 ou anterior, apenas requerem o com.amazonaws.***region***.ecr.pt Amazon ECR Parâmetro de VPC e o Amazon S3 parâmetro de avaliação gateway para tirar partido desta função.
- Amazon ECS tarefas utilizando o Fargate tipo de lançamento e plataforma versão 1.4.0 ou posterior requerem ambos os com.amazonaws.***region***.ecr.pt e com.amazonaws.***region***.ecr.api Amazon ECR Terminais de VPC, bem como a Amazon S3 parâmetro de avaliação gateway para tirar partido desta função.
- Amazon ECS tarefas utilizando o Fargate tipo de lançamento que puxa imagens dos recipientes de Amazon ECR pode restringir o acesso às VPC específicas, as tarefas utilizam e o parâmetro de avaliação VPC, o serviço utiliza-se adicionando as chaves de condição à execução da tarefa IAM função para a tarefa. Para mais informações, consulte Permissões de IAM opcionais para tarefas Fargate, abrangendo imagens ECR da Amazon sobre os parâmetros de avaliação da interface no Amazon Elastic Container Service Developer Guide.
- Amazon ECS tarefas utilizando o Fargate tipo de lançamento que puxa imagens dos recipientes de Amazon ECR que também usam o `awslogs` registar o controlador para enviar informações de registo para CloudWatch Logs requer o CloudWatch Logs Parâmetro VPC. Para obter mais informações, consulte Criar CloudWatch Logs endpoint (p. 85).
- O grupo de segurança anexado ao VPC endpoint deve permitir conexões de entrada na porta 443 na sub-rede privada da VPC.
- Atualmente, os VPC endpoints não oferecem suporte a solicitações entre regiões. Crie os VPC endpoints na mesma região da qual você planeja enviar chamadas de API para o Amazon ECR.
- Os VPC endpoints só oferecem suporte a DNS fornecido pela Amazon por meio do Amazon Route 53. Se quiser usar seu próprio DNS, poderá usar o encaminhamento de DNS condicional. Para mais informações, consulte Conjuntos de opções DHCP no Guia do usuário da Amazon VPC.
- Se os seus contentores tiverem ligações existentes para Amazon S3, as suas ligações podem ser interrompidas momentaneamente quando adicionar o Amazon S3 parâmetro de gateway. Se quiser evitar esta interrupção, crie um novo VPC que utilize o Amazon S3 terminal de gateway e depois migrar o seu Amazon ECS e os respectivos contentores para o novo VPC.

Considerações para imagens do Windows

As imagens com base no sistema operativo Windows incluem artefactos que são restringidos por licença de distribuição. Por predefinição, quando carrega imagens do Windows para um Amazon ECR repositório, as camadas que incluem estes artefactos não são empurradas à medida que são consideradas camadas estrangeiras. Quando os artefactos são fornecidos pela Microsoft, as camadas estrangeiras são obtidas a partir da infraestrutura Microsoft Azure. Por este motivo, para permitir que os seus contentores sejam

movidos para outras camadas estrangeiras de Azure, são necessárias passos adicionais para além de criar os parâmetros de avaliação de VPC.

É possível substituir este comportamento ao empurrar imagens do Windows para Amazon ECR usando o `--allow-nondistributable-artifacts` bandeira no daemon do acoplador. Quando activado, este sinalizador irá empurrar as camadas licenciadas para Amazon ECR que permite que estas imagens sejam retiradas de Amazon ECR através do terminal VPC sem acesso adicional ao Azure sendo necessário.

> **Important**
>
> Utilizar o `--allow-nondistributable-artifacts` o sinalizador não exclui a sua obrigação de cumprir os termos da licença de imagens base do contentor Windows; não pode publicar conteúdo Windows para redistribuição pública ou de terceiros. É permitida a utilização no seu próprio ambiente.

Para permitir a utilização deste sinalizador para a instalação do acoplador, tem de modificar o ficheiro de configuração daemon do acoplador que, dependendo da instalação do acoplador, pode normalmente ser configurado no menu Definições ou preferências por baixo do Motor do acoplador ou editando o `C:\ProgramData\docker\config\daemon.json` ficheiro diretamente.

Segue-se um exemplo da configuração necessária. Substitua o valor com o URI do repositório que está a carregar imagens para.

```
{
    "allow-nondistributable-artifacts": [
        "111122223333.dkr.ecr.us-west-2.amazonaws.com"
    ]
}
```

Depois de modificar o ficheiro de configuração daemon do acoplador, tem de reiniciar o daemon do acoplador antes de tentar empurrar a imagem. Confirme se o empurrão funcionou verificando se a camada base foi colocada no seu repositório.

> **Note**
>
> As camadas de base para imagens Windows são grandes. O tamanho da camada resultará num tempo mais longo para suportar e custos de armazenamento adicionais em Amazon ECR para estas imagens. Por estes motivos, recomendamos apenas utilizar esta opção quando for estritamente necessário reduzir os tempos de construção e os custos de armazenamento contínuos. Por exemplo, o `mcr.microsoft.com/windows/servercore` a imagem é aproximadamente 1,7 gib em tamanho quando comprimida Amazon ECR.

Criar os VPC endpoints para o Amazon ECR

Para criar os endpoints VPC para o Amazon ECR serviço, utilize o Criar um parâmetro de avaliação da interface procedimento no Guia do usuário da Amazon VPC.

Amazon ECS tarefas utilizando o EC2 o tipo de lançamento requer ambos Amazon ECR endpoints e o Amazon S3 parâmetro de gateway.

Amazon ECS tarefas utilizando o Fargate tipo de lançamento e plataforma, versão 1.3.0 ou anterior, apenas requerem o com.amazonaws.***region***.ecr.pt Amazon ECR Parâmetro de VPC e o Amazon S3 terminais de gateway.

Amazon ECS tarefas utilizando o Fargate tipo de lançamento e plataforma versão 1.4.0 ou posterior requerem ambos os com.amazonaws.***region***.ecr.pt e com.amazonaws.***region***.ecr.api Amazon ECR Parâmetros de avaliação de VPC e o Amazon S3 terminais de gateway.

> **Note**
>
> A ordem em que os endpoints são criados não importa.

com.amazonaws.*region*.ecr.pt

Esse endpoint é usado para as APIs de registro do Docker. Comandos do cliente do acoplador, como `push` e `pull` utilizar este parâmetro de avaliação.

Quando cria o com.amazonaws.*region*.ecr.pt o endpoint, tem de activar um nome de anfitrião DNS privado. Para o fazer, certifique-se de que Activar nome DNS privado está seleccionada na consola VPC quando criar o parâmetro VPC.

com.amazonaws.*region*.ecr.api

Note

O especificado *region* representa o identificador de região para um AWS Região suportada por Amazon ECR, como `us-east-2` para o Região do Leste dos EUA (Ohio).

Este parâmetro de avaliação é utilizado para chamadas para Amazon ECR API. As ações de API, como DescribeImages e CreateRepositories, vão para esse endpoint.

Quando o com.amazonaws.*region*.ecr.api o endpoint é criado, tem a opção de ativar um nome de anfitrião DNS privado. Activar esta definição selecionando Activar nome DNS privado na consola VPC quando criar o ponto final VPC. Se activar um nome de anfitrião de DNS privado para o parâmetro de avaliação VPC, atualize o seu SDK ou AWS CLI para a versão mais recente para que especifique um URL de ponto final ao utilizar o SDK ou AWS CLI não é necessário.

Se activar um nome de anfitrião DNS privado e estiver a utilizar um SDK ou AWS CLI versão lançada antes de 24 de janeiro de 2019, tem de utilizar o `--endpoint-url` parâmetro para especificar os parâmetros de avaliação 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 Amazon S3 parâmetro de gateway

Para o seu Amazon ECS tarefas para extrair imagens privadas de Amazon ECR, tem de criar um ponto final de gateway para Amazon S3. O endpoint de gateway é necessário porque Amazon ECR utilizações Amazon S3 para armazenar as camadas de imagens. Quando os seus contentores descarregarem imagens de Amazon ECR, têm de aceder Amazon ECR para obter o manifesto da imagem e Amazon S3 para transferir as camadas de imagens reais. Este é o nome de 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/*
```

Utilize o Criar um parâmetro de avaliação de gateway procedimento no Guia do usuário da Amazon VPC para criar o seguinte Amazon S3 parâmetro de gateway para Amazon ECR. Ao criar o endpoint, selecione as tabelas de rotas para sua VPC.

com.amazonaws.*region*s3://

O Amazon S3 o parâmetro de avaliação do gateway utiliza um IAM documento de política para limitar o acesso ao serviço. O Acesso total política pode ser usada porque as restrições que colocou na

sua tarefa IAM funções ou outros IAM as políticas do utilizador continuam a aplicar-se ao topo desta política. Se quiser limitar Amazon S3 acesso ao balde às permissões mínimas necessárias para utilização Amazon ECR, consulte Permissões mínimas do bucket do Amazon S3 para o Amazon ECR (p. 85).

Permissões mínimas do bucket do Amazon S3 para o Amazon ECR

O Amazon S3 o parâmetro de avaliação do gateway utiliza um IAM documento de política para limitar o acesso ao serviço. Para permitir apenas o mínimo Amazon S3 permissões de balde para Amazon ECR, restringir o acesso ao Amazon S3 balde que Amazon ECR quando cria o IAM documento da apólice para o parâmetro de avaliação.

A tabela a seguir descreve as permissões de política do bucket do Amazon S3 exigidas pelo Amazon ECR.

Permissão	Description (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 um AWS Região suportada por Amazon ECR, como `us-east-2` para o Região do Leste dos EUA (Ohio).

Example

O exemplo seguinte ilustra como fornecer acesso ao Amazon S3 pães necessários para Amazon ECR operações.

```
{
  "Statement": [
    {
      "Sid": "Access-to-specific-bucket-only",
      "Principal": "*",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": ["arn:aws:s3:::prod-region-starport-layer-bucket/*"]
    }
  ]
}
```

Criar CloudWatch Logs endpoint

Amazon ECS tarefas utilizando o Fargate tipo de lançamento que utilize um VPC sem um portal de Internet que também utilize o `awslogs` registar o controlador para enviar informações de registo para CloudWatch Logs exigir que crie o com.amazonaws.*region*.registos interface VPC terminal para CloudWatch Logs. Para mais informações, consulte Criar um parâmetro de avaliação de gateway no Amazon CloudWatch Logs User Guide.

Criar uma política de avaliação final para o seu Amazon ECR Parâmetros VPC

Uma política de endpoint de 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, a AWS anexará uma política padrão que permita o acesso total ao serviço. Uma política de endpoint não substitui políticas

de usuário do IAM 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 mais informações, consulte Controlo do acesso a serviços com parâmetros de avaliação de VPC no Guia do usuário da Amazon VPC.

Recomendamos a criação de um único IAM política de recursos e anexá-la a ambos Amazon ECR Parâmetros de avaliação de VPC.

Veja a seguir um exemplo de uma política de endpoint para o Amazon ECR. Essa política permite que uma função específica do IAM extraia imagens do Amazon ECR.

```
{
 "Statement": [{
  "Sid": "AllowPull",
  "Principal": {
   "AWS": "arn:aws:iam::1234567890:role/role_name"
  },
  "Action": [
   "ecr:BatchGetImage",
   "ecr:GetDownloadUrlForLayer"
  ],
  "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"
   ],
   "Resource": "*"
  }
 ]
}
```

Como modificar a política de VPC endpoint para o Amazon ECR

1. Abra o console da Amazon VPC em https://console.aws.amazon.com/vpc/.
2. No painel de navegação, escolha Parâmetros de avaliação.
3. Se você ainda não criou os VPC endpoints para o Amazon ECR, consulte Criar os VPC endpoints para o Amazon ECR (p. 83).
4. Selecione o Amazon ECR Parâmetro VPC para adicionar uma apólice e escolher o Política na metade inferior do ecrã.
5. Escolher Editar política e fazer as alterações à política.
6. Escolher Guardar para guardar a política.

Monitoramento do Amazon ECR

É possível monitorar o uso da API do Amazon ECR com o Amazon CloudWatch, que coleta e processa dados brutos do Amazon ECR em métricas legíveis, quase em tempo real. Essas estatísticas são gravadas por um período de duas semanas para que você possa acessar informações históricas e obter uma perspectiva melhor sobre o uso da API. Os dados de métricas do Amazon ECR são enviados automaticamente ao CloudWatch em períodos de um minuto. Para obter mais informações sobre o CloudWatch, consulte o Guia do usuário do Amazon CloudWatch.

O Amazon ECR fornece métricas com base no uso da API para ações de autorização, envio de imagem e extração de imagem.

O monitoramento é uma parte importante para manter a confiabilidade, disponibilidade e desempenho do Amazon ECR e das soluções da AWS. Recomendamos que você colete dados de monitoramento dos recursos que compõem sua solução da AWS para que seja possível depurar mais facilmente uma falha multipontos, caso ocorra. Porém, para 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 desempenho normal do Amazon ECR no ambiente, medindo o desempenho em vários momentos e em diferentes condições de carga. À medida que você monitora o Amazon ECR, armazene dados de monitoramento históricos para compará-los com os novos dados de desempenho, identificar padrões de desempenho normais e anomalias de desempenho, além de elaborar métodos para resolver problemas.

Tópicos

Visualização das cotas do serviço e definição de alarmes

É possível usar o console do CloudWatch para visualizar as cotas de serviço e ver como o 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 console do CloudWatch em https://console.aws.amazon.com/cloudwatch/.

2. No painel de navegação, selecione Metrics (Métricas).
3. Na guia All metrics (Todas as métricas), selecione Usage (Uso) e By AWS Resource (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 exibe o uso atual desse recurso da AWS.
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 do que, introduzir **`80`**. Isto cria um alarme que entra no estado ALARME quando a sua utilização excede 80 por cento da quota.
   c. Selecione Next (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). Depois, escolha Next (Próximo).
   e. Na próxima página, insira um nome e uma descrição para o alarme e selecione Next (Próximo).
   f. Escolha Create alarm (Criar alarme).

Métricas de uso do Amazon ECR

Você pode usar métricas de uso do CloudWatch para fornecer visibilidade sobre o uso de recursos de sua conta. Use essas métricas para visualizar o uso do serviço atual nos gráficos e painéis do CloudWatch.

As métricas de uso do Amazon ECR correspondem às cotas de serviço da AWS. 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 Amazon ECRCotas de serviço do (p. 101).

O Amazon ECR publica as métricas a seguir no namespace `AWS/Usage`.

Métrica	Description (Descrição)
`CallCount`	O número de chamadas de ação de API da sua conta. Os recursos são definidos pelas dimensões associadas à métrica.

Métrica	Description (Descrição)
	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	Description (Descrição)
`Service`	O nome do serviço da AWS 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. No momento, 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 que está sendo acompanhado. No momento, o Amazon ECR não usa a dimensão de classe.

Relatórios de uso do Amazon ECR

A AWS fornece uma ferramenta de geração de relatório 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 você pode 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 obter mais informações, consulte Marcar recursos para faturamento (p. 28).

Para obter mais informações sobre como criar um Relatório de custo e uso do AWS, consulte Relatório de custo e uso da AWS no Guia do usuário do AWS Billing and Cost Management.

Amazon ECR eventos e EventBridge

O Amazon EventBridge permite que você automatize os serviços da AWS e responda automaticamente aos eventos do sistema, como problemas de disponibilidade do aplicativo ou alterações de recursos. Os eventos dos serviços da AWS são entregues ao EventBridge quase 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. As ações que podem ser automaticamente acionadas incluem as seguintes:

- Adicionar eventos a grupos de logs no CloudWatch Logs
- Como invocar uma função do AWS Lambda
- Invocação do Run Command do Amazon EC2
- Retransmissão do evento para o Amazon Kinesis Data Streams
- Ativação da máquina de estado do AWS Step Functions
- Notificação de um tópico do Amazon SNS ou de uma fila do AWS SMS

Para obter mais informações, consulte Conceitos básicos do Amazon EventBridge no Guia do usuário do Amazon EventBridge.

Eventos de amostras de Amazon ECR

Veja a seguir exemplos de eventos do Amazon ECR.

Ocorrência para um envio de imagem concluído

O evento a seguir é enviado quando cada envio de imagem é concluído. Para obter mais informações, consulte Enviar uma imagem (p. 30).

```
{
    "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-repo",
        "image-digest":
 "sha256:7f5b2640fe6fb4f46592dfd3410c4a79dac4f89e4782432e0378abcd1234",
        "action-type": "PUSH",
        "image-tag": "latest"
    }
}
```

Evento para uma digitalização de imagem concluída

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 obter mais informações, consulte Verificação de imagens (p. 50).

```
{
```

```
    "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-repo"
    ],
    "detail": {
        "scan-status": "COMPLETE",
        "repository-name": "my-repo",
        "finding-severity-counts": {
        "CRITICAL": 10,
        "MEDIUM": 9
      },
        "image-digest":
 "sha256:7f5b2640fe6fb4f46592dfd3410c4a79dac4f89e4782432e0378abcd1234",
        "image-tags": []
    }
}
```

Evento para eliminação de uma imagem

O evento a seguir é enviado quando uma imagem é excluída. Para obter mais informações, consulte Excluir uma imagem (p. 35).

```
{
    "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-repo",
        "image-digest":
 "sha256:7f5b2640fe6fb4f46592dfd3410c4a79dac4f89e4782432e0378abcd1234",
        "action-type": "DELETE",
        "image-tag": "latest"
    }
}
```

Início de sessão Amazon ECR ações com AWS CloudTrail

O Amazon ECR é integrado ao AWS CloudTrail, um serviço que fornece um registro das ações executadas por um usuário, uma função ou um serviço da AWS no Amazon ECR. O 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 realizadas devido às definições de encriptação nos seus repositórios
- Todas as ações tomadas devido às regras de política de ciclo de vida, incluindo ações bem-sucedidas e malsucedidas

Quando uma trilha é criada, você pode habilitar a entrega contínua de eventos do CloudTrail para um bucket do Amazon S3 incluindo eventos do Amazon ECR. Se não configurar uma trilha, você ainda poderá visualizar os eventos mais recentes no console do CloudTrail em Event history. Com essa informação, você pode determinar a solicitação que foi feita ao Amazon ECR, o endereço IP de origem, de onde a solicitação partiu quando foi feita, e detalhes adicionais.

Para obter mais informações, consulte AWS CloudTrail User Guide.

Informações sobre o Amazon ECR no CloudTrail

O CloudTrail está habilitado na sua conta da AWS ao criá-la. Quando uma atividade ocorrer no Amazon ECR, ela será registrada em um evento do CloudTrail com outros eventos de serviços da AWS em Event history (Histórico de eventos). Você pode visualizar, pesquisar e fazer download de eventos recentes em sua conta da AWS. Para obter mais informações, consulte Visualizar eventos com o histórico de eventos do CloudTrail.

Para obter um registro contínuo de eventos em sua conta da AWS, incluindo eventos para o Amazon ECR, crie uma trilha. Uma trilha permite CloudTrail para fornecer arquivos de log a 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 partição da AWS e fornece os arquivos de log para o bucket do Amazon S3 que você especificar. Além disso, é possível configurar outros serviços da AWS para analisar e agir sobre os dados de evento coletados nos logs do CloudTrail. Para obter mais informações, consulte:

- Criar uma trilha para a conta da AWS
- Integrações de serviços da AWS com logs do CloudTrail
- Configurar notificações SNS Amazon para CloudTrail
- Receber arquivos de log do CloudTrail de várias regiões e receber arquivos de log do CloudTrail de várias contas

Todas as ações de API do Amazon ECR são registradas em log pelo CloudTrail e documentadas no Amazon Elastic Container Registry API Reference. Quando você executa tarefas comuns, são geradas seções nos arquivos de log do CloudTrail para cada ação de API que faz parte dessa tarefa. Por exemplo, quando você cria um repositório, são geradas as seções `GetAuthorizationToken`, `CreateRepository` e `SetRepositoryPolicy` nos arquivos de log do CloudTrail. 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 registo (p. 94).

Cada evento ou entrada no log 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 IAM.
- Se a solicitação foi feita com credenciais de segurança temporárias de uma função ou de um usuário federado
- Se a solicitação foi feita por outro serviço da AWS

Para obter mais informações, consulte Elemento do CloudTrail `userIdentity`.

Noções básicas das entradas dos arquivos de log do Amazon ECR

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log em um bucket do Amazon S3 que você especificar. Os arquivos de log do CloudTrail contêm uma ou mais entradas de

log. Um evento representa uma única solicitação de qualquer origem e inclui informações sobre a ação solicitada, a data e hora da ação, parâmetros de solicitação, além de outras informações. Os arquivos de log do CloudTrail não são um rastreamento de pilha ordenada das chamadas da API pública, portanto, não são exibidos em nenhuma ordem específica.

CloudTrail exemplos de entrada de registo

Veja a seguir exemplos de entradas de log do CloudTrail para algumas tarefas comuns do Amazon ECR.

> **Note**
>
> Estes exemplos foram formatados para obter melhor legibilidade. Em um arquivo de log do CloudTrail, 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 log real do CloudTrail, você vê entradas e eventos de vários serviços da AWS.

Tópicos

Exemplo: Criar ação de repositório

O exemplo a seguir mostra uma entrada de log do CloudTrail que demonstra a ação `CreateRepository`.


```
{
    "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: AWS KMS CriarSubvencionar ação API ao criar uma Amazon ECR repositório

O exemplo seguinte mostra um CloudTrail que demonstre a AWS KMS Criaração de Subvenção ao criar uma Amazon ECR com encriptação KMS ativada. Para cada repositório que é criado com encriptação KMS está ativada, deve ver duas entradas de registo CreateGrant em 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 push de imagem

O exemplo a seguir mostra uma entrada de log do CloudTrail que demonstra o envio de uma imagem, que usa a ação `PutImage`.

> **Note**
>
> Ao enviar uma imagem, você também verá as referências `InitiateLayerUpload`, `UploadLayerPart` e `CompleteLayerUpload` nos logs do CloudTrail.


```
{
    "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 da imagem

O exemplo a seguir mostra uma entrada de log do CloudTrail que demonstra a extração de uma imagem, que usa a ação `BatchGetImage`.

> **Note**
>
> Ao extrair uma imagem, se você ainda não tiver a imagem armazenada localmente, também serão exibidas as referências `GetDownloadUrlForLayer` nos logs do CloudTrail.


```
{
    "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 política do ciclo de vida das imagens

O exemplo a seguir mostra uma entrada de log do CloudTrail que demonstra quando uma imagem expirou 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.

```
{
    "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
            }
        ]
    }
}
```

Amazon ECRCotas de serviço do

A tabela a seguir fornece as cotas de serviço padrão para o Amazon Elastic Container Registry (Amazon ECR).

Cota de serviço	Description (Descrição)	Valor de cota padrão
Repositórios registrados	O número máximo de repositórios que você pode criar por Região.	10.000*
Imagem por repositório	O número máximo de imagens por repositório.	10.000*

A tabela a seguir fornece as cotas de taxa padrão para cada uma das ações de API do Amazon ECR envolvidas com as ações de envio de imagem e de extração de imagem.

Amazon ECRAção do	Operação de API	Description (Descrição)	Valor de cota padrão
Autenticação	Taxa de solicitações de GetAuthorizationToken	A taxa de solicitações da API GetAuthorizationToken que podem ser feitas por segundo, por região.	200 USD
Envio de imagem	Taxa de solicitações de BatchCheckLayerAvailability	A taxa de solicitações da API BatchCheckLayerAvailability que podem ser feitas por segundo, por região. 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.	200 USD
	Taxa de solicitações de InitiateLayerUpload	A taxa de solicitações da API InitiateLayerUpload que podem ser feitas por segundo, por região. Quando uma imagem é enviada, a API InitiateLayerUpload é chamada uma vez por camada de imagem da	10*

Amazon ECRAção do	Operação de API	Description (Descrição)	Valor de cota padrão
		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.	
	Taxa de solicitações de CompleteLayerUpload	A taxa de solicitações da API CompleteLayerUpload que podem ser feitas por segundo, por região. 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.	10*
	Taxa de solicitações de UploadLayerPart	A taxa de solicitações da API UploadLayerPart que podem ser feitas por segundo, por região. Quando uma imagem é enviada, é feito upload de cada nova camada da imagem em partes. O tamanho máximo de cada parte da camada da imagem pode ser de 20.971.520 bytes (ou cerca de 20 MB). A API UploadLayerPart é chamada uma vez por cada nova parte da camada da imagem.	260
	Taxa de solicitações de PutImage	A taxa de solicitações da API PutImage que podem ser feitas por segundo, por região. 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.	10*

Amazon ECRAção do	Operação de API	Description (Descrição)	Valor de cota padrão
Extração de imagem	Taxa de solicitações de BatchGetImage	A taxa de solicitações da API BatchGetImage que podem ser feitas por segundo, por região. Quando uma imagem é extraída, a API BatchGetImage é chamada uma vez para recuperar o manifesto da imagem.	1.000.
	Taxa de solicitações de GetDownloadUrlForLayer	A taxa de solicitações da API GetDownloadUrlForLayer que podem ser feitas por segundo, por região. Quando uma imagem é extraída, a API GetDownloadUrlForLayer é chamada uma vez por camada de imagem que ainda não está armazenada em cache.	1,500

A tabela a seguir mostra outras cotas para o Amazon ECR e as imagens de Docker que não podem ser alteradas.

Note

As informações da parte da camada mencionadas na tabela a seguir são aplicáveis somente se você estiver chamando as ações de API do Amazon ECR diretamente para iniciar multipart uploads para operações de envio de imagens. Essa é uma ação rara. Recomendamos usar a CLI do Docker para extrair, marcar e enviar imagens.

Cota de serviço	Description (Descrição)	Valor da cota
Partes da camada	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.	1.000.
Tamanho máximo da camada	O tamanho máximo (MiB) de uma camada.**	10.000*
Tamanho mínimo da parte da camada	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	5.

Cota de serviço	Description (Descrição)	Valor da cota
	multipart uploads para operações de envio de imagem.	
Tamanho máximo da parte da camada	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 para operações de envio de imagem.	10*
Tags por imagem	O número máximo de tags por imagem	1000
Duração da política de ciclo de vida	O número máximo de caracteres em uma política de ciclo de vida.	30.720
Regras por política de ciclo de vida	O número máximo de regras em uma política de ciclo de vida	50
Taxa de verificações de imagens	O número máximo de verificações de imagens por imagem, por dia.	1

** O tamanho máximo da camada listado aqui é calculado multiplicando o tamanho máximo da parte da camada (10 MiB) pelo número máximo das partes da camada (1.000).

Gerir o seu Amazon ECR quotas de serviço no Console de gerenciamento da AWS

Amazon ECR integrado com Cotas de serviço, um AWS serviço que lhe permite ver e gerir as suas quotas a partir de uma localização central. Para mais informações, consulte O que são quotas de serviço? no Guia do usuário do Cotas de serviço.

Cotas de serviçoO Amazon ECR facilita a pesquisa do valor de todas as cotas de serviço do .

Para ver Amazon ECR quotas de serviço (Console de gerenciamento da AWS)

1. Abra o console do Cotas de serviço em https://console.aws.amazon.com/servicequotas/.
2. No painel de navegação, escolha Serviços AWS.
3. Desde o AWS serviços lista, pesquisar e selecionar Amazon Elastic Container Registry (Amazon ECR).

   No Quotas de serviço lista, pode ver o nome da quota de serviço, o valor aplicado (se estiver disponível), AWS quota predefinida e se o valor da quota é ajustável.
4. Para visualizar informações adicionais sobre uma cota de serviço, como a descrição, escolha o nome da cota.

Para solicitar um aumento da quota, consulte Solicitar um aumento da quota no Guia do usuário do Cotas de serviço.

Como criar um alarme do CloudWatch para monitorar métricas de uso da API

Amazon ECR fornece CloudWatch métricas de utilização que correspondem ao AWS quotas de serviço para cada uma das API envolvidas na autenticação do registo, no push de imagens e nas ações de extração de imagens. No console do Cotas de serviço, é possível visualizar o uso em um gráfico e configurar alarmes que o alertarão quando o uso se aproximar de uma cota de serviço. Para obter mais informações, consulte Métricas de uso do Amazon ECR (p. 89).

Utilize os passos seguintes para criar uma CloudWatch com base num dos Amazon ECR Métricas de utilização de API.

Para criar um alarme com base no seu Amazon ECR quotas de utilização (Console de gerenciamento da AWS)

1. Abra o console do Cotas de serviço em https://console.aws.amazon.com/servicequotas/.
2. No painel de navegação, escolha Serviços AWS.
3. Desde o AWS serviços lista, pesquisar e selecionar Amazon Elastic Container Registry (Amazon ECR).
4. No Quotas de serviço lista, selecione Amazon ECR quota de utilização para a qual pretende criar um alarme para.
5. No Eventos do Amazon CloudWatch secção de alarmes, escolher Criar.
6. Para Limite de alarme, escolha a percentagem do valor da quota aplicada que pretende definir como o valor do alarme.
7. Para Nome do alarme, introduza um nome para o alarme e escolha Criar.

Amazon ECRSolução de problemas com o

Este capítulo ajuda a encontrar informações de diagnóstico para o Amazon Elastic Container Registry (Amazon ECR) e mostra as etapas de solução de problemas e mensagens de erro comuns.

Tópicos

Como habilitar a saída de depuração do Docker

Para começar a depurar qualquer problema relacionado ao Docker, você deve começar ativando a saída de depuração do Docker no daemon do Docker em execução nas instâncias do host. Para obter mais informações sobre como permitir a depuração do acoplador, se estiver a utilizar imagens retiradas de Amazon ECR em Amazon ECS instâncias de recipiente, ver Ativar saída de depuração do acoplador no Amazon Elastic Container Service Developer Guide.

Ativar o AWS CloudTrail

Informações adicionais sobre erros devolvidos por Amazon ECR pode ser descoberto ao permitir AWS CloudTrail, que é um serviço que regista AWS chamadas para a sua conta AWS. CloudTrail fornece ficheiros de registo a um Amazon S3 balde. Ao usar informações coletadas pelo CloudTrail, você pode determinar quais solicitações foram feitas para os serviços da AWS, quem fez a solicitação, quando ela foi feita etc. Para saber mais sobre o CloudTrail, incluindo como ativá-lo e encontrar seus arquivos de log, consulte o AWS CloudTrail User Guide. Para mais informações sobre a utilização CloudTrail com Amazon ECR, consulte Início de sessão Amazon ECR ações com AWS CloudTrail (p. 92).

Como otimizar o desempenho para o Amazon ECR

A seção a seguir fornece recomendações sobre configurações e estratégias que podem ser utilizadas 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 uploads simultâneos ao Amazon ECR, o que resulta em tempos de upload 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 rascunho do Docker. Para mais informações, consulte Criar uma imagem de base na documentação do acoplador.

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 ao extrair imagens do Amazon ECR ao usar 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 poderá 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 transferida para AWS CLI comandos utilizando o --region parâmetro ou definido como região predefinida para um perfil utilizando o aws configure comando. Você também pode definir a região ao fazer chamadas usando o SDK da AWS. Para obter mais informações, consulte a documentação do SDK para a sua linguagem de programação específica.

Solução de problemas de erros com comandos do Docker ao usar o Amazon ECR

Tópicos

Em alguns casos, a execução de 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.

Erro "Filesystem Verification Failed" ou "404: Imagem não encontrada" ao extrair uma imagem de um Amazon ECR Repositório

Pode receber o erro `Filesystem verification failed` ao utilizar o docker pull para puxar uma imagem de um Amazon ECR repositório com acoplador 1.9 ou superior. 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 à Internet. Verifique suas configurações de rede e se outros aplicativos e ferramentas podem acessar recursos na Internet. Se estiver a correr docker pull num Amazon EC2 exemplo numa subrede privada, verifique se a subrede tem uma via 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 meio do seu firewall corporativo para o Amazon Simple Storage Service (Amazon S3). Se sua organização usa o software de firewall ou um dispositivo NAT que permite endpoints de serviço do Amazon S3, verifique se os endpoints de serviço 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 mais informações, consulte Proxy HTTP na documentação do acoplador.

Erro "Sistema de ficheiroVerificação da camada falhou" ao puxar imagens de Amazon ECR

Pode receber o erro `image image-name not found` ao puxar imagens utilizando o docker pull comando. 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 completo ou um problema de rede pode impedir a transferência de uma ou mais camadas, conforme discutido anteriormente sobre o `Filesystem verification failed` mensagem. 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 HTTP 403 ou o erro "Não há credenciais de autenticação básica" ao enviar ao repositório

Existem alturas em que pode receber um `HTTP 403 (Forbidden)` erro ou mensagem de erro `no basic auth credentials` do docker push ou docker pull comandos, mesmo que tenha autenticado com sucesso o acoplador utilizando o aws ecr get-login-password comando. 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.

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 obter mais informações, consulte Políticas de repositório (p. 20).

Seu token expirou

O período de expiração padrão para tokens de autorização obtidos usando a operação `GetAuthorizationToken` é de 12 horas.

Insetos em `wincred` gestor de credenciais

Algumas versões do acoplador para Windows usam um gestor de credenciais chamado `wincred`, que não manuseia correctamente o comando de início de sessão do acoplador produzido por aws ecr get-login (para 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. Pode contornar este erro removendo o `https://` do

argumento do registo no comando de início de sessão do acoplador que é saída de aws ecr get-login. É apresentado abaixo um comando de início de sessão do acoplador sem o esquema HTTPS.

```
docker login -u AWS -p <password> <aws_account_id>.dkr.ecr.<region>.amazonaws.com
```

Resolução de problemas Amazon ECR Mensagens de erro

Em alguns casos, uma chamada API que tenha desencadeado através do Amazon ECS consola ou AWS CLI sai com uma mensagem de erro. Algumas mensagens de erro comuns e possíveis soluções são explicadas abaixo.

Erro "Resposta de erro do Daemon: Parâmetro de Avaliação do Registo Inválido" Quando Executar o início de sessão do ecr

Pode ver o seguinte erro ao executar o aws ecr get-login para obter as credenciais de início de sessão para o seu Amazon ECR repositório:

```
Error response from daemon: invalid registry endpoint
    https://xxxxxxxxxxxx.dkr.ecr.us-east-1.amazonaws.com/v0/: unable to ping registry
 endpoint
    https://xxxxxxxxxxxx.dkr.ecr.us-east-1.amazonaws.com/v0/
v2 ping attempt failed with error: Get https://xxxxxxxxxxxx.dkr.ecr.us-
east-1.amazonaws.com/v2/:
    dial tcp: lookup xxxxxxxxxxxx.dkr.ecr.us-east-1.amazonaws.com on 172.20.10.1:53:
    read udp 172.20.10.1:53: i/o timeout
```

Esse erro pode ocorrer em sistemas MacOS X e Windows que executam a caixa de ferramentas do Docker, o Docker para Windows ou o Docker para Mac. É frequentemente causado quando outras aplicações alteram as rotas através do gateway local (192.168.0.1) através do qual a máquina virtual tem de ligar para aceder ao Amazon ECR serviço. Se o erro ocorrer durante o uso da caixa ferramentas do Docker, isso poderá ser resolvido com frequência com a reinicialização do ambiente de máquina do Docker ou do sistema operacional local do cliente. Se isso não solucionar o problema, use o comando docker-machine ssh para entrar em sua instância de contêiner. Realize uma busca do DNS em um host externo para verificar se vê os mesmos resultados que aparecem em seu host local. Se os resultados forem diferentes, consulte a documentação da caixa ferramentas do Docker para verificar se seu ambiente de máquina do Docker está configurado corretamente.

HTTP 429: Demasiados pedidos ou exceção

Pode receber um `429: Too Many Requests` erro ou `ThrottleException` erro de um ou mais Amazon ECR comandos ou chamadas API. Se estiver a utilizar ferramentas do acoplador com Amazon ECR, então para versões do acoplador 1.12.0 e superior, pode ver a mensagem de erro `TOOMANYREQUESTS: Rate exceeded`. Para versões do acoplador abaixo de 1.12.0, pode ver o erro `Unknown: Rate exceeded`.

Isso indica que você está chamando um único endpoint no Amazon ECR repetidamente em um curto intervalo e que suas solicitações estão sendo suspensas. A suspensão ocorre quando as chamadas para um único endpoint de um único usuário ultrapassam um determinado limite em um período.

Várias operações de API no Amazon ECR têm suspensões diferentes.

Por exemplo, o acelerador para o `GetAuthorizationToken` a acção é de 20 transacção por segundo (TPS), com um disparo de até 200 TPS permitido. Em cada região, cada conta recebe um balde que pode armazenar até 200 `GetAuthorizationToken` créditos. 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 gerenciar os erros de suspensão, implemente uma função de novas tentativas com backoff adicional no código. Para mais informações, consulte Retries de erros e Backoff Exponencial em AWS no Referência geral dos serviços web Amazon.

HTTP 403: "O utilizador [arn] não está autorizado a realizar [operação]"

Você poderá receber o seguinte erro ao tentar realizar uma ação com o Amazon ECR:

```
$ aws ecr get-login
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 concedidas 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 Identity and Access Management para o Amazon Elastic Container Registry (p. 60).

HTTP 404: Erro "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 possam ser 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 Amazon ECR repositórios (p. 17).

Solução de problemas de verificação de imagem

Veja a seguir falhas comuns de verificação de imagem. Pode ver erros como este no Amazon ECR apresentando os detalhes da imagem ou através da API ou AWS CLI usando o `DescribeImageScanFindings` API.

UnsupportedImageError

> Pode obter um `UnsupportedImageError` erro ao tentar digitalizar uma imagem que foi construída utilizando um sistema operativo que Amazon ECR não suporta a digitalização de imagens para. Amazon ECR suporta a análise de vulnerabilidade de pacotes para as principais versões do Amazon Linux, Amazon Linux 2, Distribuições de Debian, Ubuntu, centos, Oracle Linux, Alpine e RHEL Linux. Quando uma distribuição perder o apoio do seu fornecedor, Amazon ECR já não pode suportar a leitura para vulnerabilidades. Amazon ECR não suporta imagens de digitalização criadas a partir do Espigão do acoplador imagem.

Um nível de severidade `UNDEFINED` é retornado

Pode receber um exame de digitalização que tenha um nível de gravidade de `UNDEFINED`. Seguem-se as causas comuns para isto:

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

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

Histórico de documentos

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	Description (Descrição)	Date
Suporte de artefactos OCI	Amazon ECR adicionado apoio para empurrar e puxar artefactos da Iniciativa de Contentores Abertos (Open Container Initiative, OCI). Um novo parâmetro `artifactMediaType` foi adicionado ao `DescribeImages` Resposta API para indicar o tipo de artefacto. Para obter mais informações, consulte Empurrar um gráfico Helm (p. 32).	24 de agosto de 2020
Criptografia em repouso	Amazon ECR adicionado suporte para configurar encriptação para os seus repositórios utilizando encriptação lado do servidor com chaves mestre de clientes (cmks) armazenadas em AWS Key Management Service (AWS KMS). Para obter mais informações, consulte Criptografia em repouso (p. 75).	29 de julho de 2020
Imagens multiarquitetura	Amazon ECRO adicionou suporte à criação e ao envio de listas de manifesto do Docker usadas para imagens multiarquitetura. Para obter mais informações, consulte Enviar uma imagem multiarquitetura (p. 31).	28 de abril de 2020
Amazon ECRMétricas de uso do	Amazon ECR adicionado CloudWatch métricas de utilização que fornecem visibilidade na utilização de recursos da sua conta. Também tem a capacidade de criar CloudWatch alarmes de ambos os CloudWatch e Cotas de serviço consolas para obter alertas quando a sua utilização se aproxima da sua quota de serviço aplicada. Para obter mais informações, consulte Métricas de uso do Amazon ECR (p. 89).	28 de fevereiro de 2020
Atualizado Amazon ECR quotas de serviço	Atualização das cotas de serviço do Amazon ECR para incluir cotas por API. Para obter mais informações, consulte Amazon ECRCotas de serviço do (p. 101).	19 de fevereiro de 2020
Adicionado `get-login-password` comando	Adição de suporte para get-login-password, que oferece um método simples e seguro para recuperar um token de autorização. Para obter mais informações, consulte Utilizar um token de autorização (p. 14).	4 de fevereiro de 2020

Alteração	Description (Descrição)	Date
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 de descobertas da verificação. Para obter mais informações, consulte Verificação de imagens (p. 50).	24 de outubro de 2019
Política de VPC endpoint	Adicionado suporte para definir um IAM política sobre Amazon ECR interface de parâmetros de avaliação de VPC. Para obter mais informações, consulte Criar uma política de avaliação final para o seu Amazon ECR Parâmetros VPC (p. 85).	26 de setembro de 2019
Mutabilidade de tag de imagem	Adição de suporte à configuração de um repositório para ser imutável a fim de impedir que as tags de imagem sejam substituídas. Para obter mais informações, consulte Mutabilidade de tag de imagem (p. 49).	25 de julho de 2019
VPC endpoints de interface (AWS PrivateLink)	Adicionado suporte para a configuração de VPC endpoints de interface desenvolvidos pelo AWS PrivateLink. Isso permite criar uma conexão privada entre sua VPC e o Amazon ECR sem exigir acesso pela Internet por meio de uma instância NAT, de uma conexão VPN ou do AWS Direct Connect. Para obter mais informações, consulte Amazon ECR configurar os parâmetros de avaliação VPC (AWS privaçãoligação) (p. 81).	25 de janeiro de 2019
Marcação de recursos	Amazon ECRO adicionou suporte para a adição de tags de metadados aos seus repositórios. Para obter mais informações, consulte Marcar um Amazon ECR repositório (p. 26).	18 de dezembro de 2018
Amazon ECRAlteração do nome do	Amazon Elastic Container Registry é renomeado (anteriormente Amazon EC2 Registo de Contentores).	21 de novembro de 2017
Políticas de ciclo de vida	Amazon ECRAs políticas de ciclo de vida do permitem que você especifique o gerenciamento do ciclo de vida das imagens em um repositório. Para obter mais informações, consulte Políticas de ciclo de vida (p. 37).	11 de outubro de 2017

Alteração	Description (Descrição)	Date
Amazon ECRSuporte do à imagem de docker manifesto 2, esquema 2	Amazon ECRO agora oferece suporte a esquema 2 de manifesto V2 de imagem de docker (usado com o Docker versão 1.10 e posteriores). Para obter mais informações, consulte Formatos de manifesto de imagem de contêiner (p. 54).	27 de janeiro de 2017
Amazon ECRDisponibilidade geral do	Amazon Elastic Container Registry (Amazon ECR)O AWS é um serviço gerenciado de registro seguro, dimensionável e confiável do Docker da .	21 de dezembro de 2015

AWS Glossary

For the latest AWS terminology, see the AWS Glossary in the AWS General Reference.

Se fornecermos uma tradução da versão em inglês do guia, a versão em inglês prevalecerá caso haja qualquer conflito entre as versões. A tradução é fornecida com o uso de tradução por máquina.