aws

Whitepaper da AWS

Diretrizes de melhores práticas e padrões de design: otimização da performance do Amazon S3

Diretrizes de melhores práticas e padrões de design: otimização da performance do Amazon S3: Whitepaper da AWS

Table of Contents

Diretrizes de melhores práticas e padrões de design: otimização da performance do Amazon S3

Data de publicação inicial: junho de 2019 (Revisões do documento)

Resumo

Ao construir aplicações que carregam e recuperam armazenamento do Amazon S3, siga as diretrizes de práticas recomendadas da AWS para otimizar a performance. A AWS também oferece Padrões de design de performance mais detalhados.

Introdução

As aplicações podem executar facilmente milhares de transações por segundo em performance de solicitação ao fazer upload e recuperar armazenamento do Amazon S3. O Amazon S3 escala automaticamente para taxas de solicitações elevadas. Por exemplo, a aplicação pode atingir pelo menos solicitações 3.500 `PUT/COPY/POST/DELETE` e 5.500 `GET/HEAD` por segundo por prefixo em um bucket. Não há limite para o número de prefixos em um bucket. Você pode aumentar seu desempenho de leitura ou gravação paralelizando as leituras. Por exemplo, se você criar 10 prefixos em um bucket do Amazon S3 para paralelizar leituras, poderá escalar o desempenho de leitura para 55.000 solicitações de leitura por segundo.

Por exemplo, algumas aplicações de data lake no Amazon S3 verificam vários milhões ou bilhões de objetos para consultas que são executadas em petabytes de dados. Essas aplicações de data lake atingem taxas de transferência de instância única que maximizam o uso da interface de rede para a instância do Amazon EC2, que podem ser de até 100 Gb/s em uma única instância. Esses aplicativos então agregam a taxa de transferência em várias instâncias para obter vários terabits por segundo.

Outros aplicativos são sensíveis à latência, como aplicativos de mensagem de mídias sociais. Essas aplicações podem atingir latências consistentes de objetos pequenos (e latências de saída de primeiro byte para objetos maiores) de aproximadamente 100 a 200 milissegundos.

Outros serviços da AWS também podem ajudar a acelerar o desempenho para diferentes arquiteturas de aplicativo. Por exemplo, para obter taxas de transferência mais altas em uma única conexão HTTP ou latências de um dígito de milissegundos, use o Amazon CloudFront ou o Amazon ElastiCache para armazenar em cache com o Amazon S3.

Além disso, se você quiser um transporte de dados rápido em longas distâncias entre um cliente e um bucket do S3, use o Amazon S3 Transfer Acceleration. O Transfer Acceleration usa os pontos de presença distribuídos globalmente no CloudFront para acelerar o transporte de dados em distâncias geográficas.

Se a workload do Amazon S3 usa criptografia no lado do servidor com o AWS Key Management Service (SSE-KMS), consulte Limites do AWS KMS no Guia do desenvolvedor do AWS Key Management Service para obter informações sobre as taxas de solicitações compatíveis com seu caso de uso.

Os tópicos a seguir descrevem as diretrizes de melhores práticas e os padrões de design para otimizar a performance de aplicações que usam o Amazon S3.

Essas diretrizes prevalecem sobre as diretrizes anteriores relacionadas à otimização da performance do Amazon S3. Por exemplo, as diretrizes anteriores de performance do Amazon S3 recomendavam a randomização da nomenclatura de prefixos com caracteres com hash para otimizar a performance de recuperações de dados frequentes. Não é mais necessário randomizar a nomeação de prefixos para o desempenho, e você pode usar nomeação sequencial baseada em datas para seus prefixos. Consulte as Diretrizes de performance e os Padrões de design de performance para obter as informações mais recentes sobre a otimização de performance para o Amazon S3.

Diretrizes de performance do Amazon S3

A fim de obter a melhor performance para a aplicação no Amazon S3, a AWS recomenda as diretrizes a seguir.

Tópicos

Avaliar o desempenho

Ao otimizar a performance, observe os requisitos de taxa de transferência de rede, CPU e DRAM (memória dinâmica de acesso aleatório). Dependendo da combinação de demandas desses recursos diferentes, convém avaliar os diferentes tipos de instância do Amazon EC2. Para obter mais informações sobre tipos de instância, consulte Tipos de instância no Guia do usuário do Amazon EC2 para instâncias do Linux.

Também é útil observar o tempo de pesquisa de DNS, a latência e a velocidade de transferência dos dados usando ferramentas de análise de HTTP ao avaliar o desempenho.

Dimensionar conexões de armazenamento na horizontal

A distribuição de solicitações em muitas conexões é um padrão de design comum para dimensionamento horizontal do desempenho. Ao criar aplicações de alta performance, pense no Amazon S3 como um sistema distribuído muito grande, não como um único endpoint de rede de um servidor de armazenamento tradicional. Para atingir a melhor performance, emita várias solicitações simultâneas para o Amazon S3. Espalhe essas solicitações em conexões separadas para maximizar

a largura de banda acessível do Amazon S3. O Amazon S3 não tem limites para o número de conexões feitas ao bucket.

Usar consulta na escala de bytes

Usando o cabeçalho HTTP Range em uma solicitação GET Object, é possível obter um objeto em escala de bytes, transferindo somente a parte especificada. É possível usar conexões simultâneas ao Amazon S3 para buscar diferentes escalas de bytes no mesmo objeto. Isso ajuda a atingir um throughput agregado maior em comparação com uma única solicitação de objeto inteiro. A consulta de escalas menores de um objeto grande também permite que o aplicativo melhore os tempos de repetição quando as solicitações são interrompidas. Para obter mais informações, consulte Obter objetos.

Os tamanhos típicos das solicitações de escala de bytes são 8 MB ou 16 MB. Se os objetos usarem a solicitação PUT com um upload de várias partes, é recomendado usar a solicitação GET nos mesmos tamanhos de parte (ou pelo menos de acordo com os limites de parte) para obter o melhor desempenho. As solicitações GET podem abordar partes individuais diretamente; por exemplo, `GET ?partNumber=N`.

Solicitações de repetição para aplicativos sensíveis à latência

Repetições e tempos limite agressivos ajudam a obter uma latência consistente. Devido à grande escala do Amazon S3, se a primeira solicitação for lenta, uma solicitação repetida provavelmente seguirá um caminho diferente e será bem-sucedida rapidamente. Os AWS SDKs têm valores configuráveis de tempo limite e repetição que podem ser ajustados de acordo com as tolerâncias do aplicativo específico.

Combinar o Amazon S3 (armazenamento) e o Amazon EC2 (computação) na mesma região da AWS

Embora os nomes de buckets do S3 sejam globalmente exclusivos, cada bucket é armazenado em uma região selecionada ao criar o bucket. Para otimizar a performance, recomendamos acessar o bucket nas instâncias do Amazon EC2 na mesma região da AWS quando possível. Isso ajuda a reduzir os custos de latência de rede e transferência de dados.

Para obter mais informações sobre custos de transferência de dados, consulte Definição de preço do Amazon S3.

Usar o Amazon S3 Transfer Acceleration para minimizar a latência causada pela distância

O Amazon S3 Transfer Acceleration gerencia transferências de arquivos rápidas, fáceis e seguras em longas distâncias entre o cliente e um bucket do S3. O Transfer Acceleration tira proveito dos pontos de presença distribuídos globalmente no Amazon CloudFront. Conforme os dados chegam a um ponto de presença, eles são roteados para o Amazon S3 por um caminho de rede otimizado. O Transfer Acceleration é ideal para transferir gigabytes a terabytes de dados regularmente entre os continentes. Ele também é útil para clientes que fazem upload em um bucket centralizado do mundo todo.

Você pode usar a Ferramenta de comparação de velocidade do Amazon S3 Transfer Acceleration para comparar velocidades de upload aceleradas e não aceleradas em regiões do Amazon S3. A ferramenta de comparação de velocidade usa multipart uploads para transferir um arquivo do navegador para várias regiões do Amazon S3 com e sem o uso do Amazon S3 Transfer Acceleration.

Usar a versão mais recente dos AWS SDKs

Os AWS SDKs têm compatibilidade incorporada com muitas da diretrizes recomendadas para otimizar a performance do Amazon S3. Os SDKs fornecem uma API mais simples para aproveitar o Amazon S3 em uma aplicação e são atualizados regularmente para seguir as práticas recomendadas mais recentes. Por exemplo, os SDKs incluem uma lógica para executar automaticamente solicitações de repetição em erros HTTP 503 e estão investindo em código para responder e se adaptar a conexões lentas.

Os SDKs também fornecem o Gerenciador de transferências, que automatiza conexões de dimensionamento horizontal para atingir milhares de solicitações por segundo, usando solicitações na escala de bytes quando apropriado. É importante usar a última versão dos AWS SDKs para obter os recursos mais recentes de otimização de desempenho.

Também é possível otimizar o desempenho ao usar solicitações de API REST HTTP. Ao usar a API REST, siga as mesmas práticas recomendadas que fazem parte dos SDKs. Permita tempo limite e repetição em solicitações lentas e que várias conexões consultem dados de objeto em paralelo. Para obter informações sobre como usar a API REST, consulte a Referência da API do Amazon Simple Storage Service.

Padrões de design de performance do Amazon S3

Ao projetar aplicações para carregar e recuperar armazenamento do Amazon S3, use nossas práticas recomendadas e padrões de design para atingir a melhor performance da aplicação. Também oferecemos as Diretrizes de performance para você considerar ao planejar a arquitetura da aplicação.

Para otimizar o desempenho, você pode usar os padrões de design a seguir.

Tópicos

Usar o cache para conteúdo acessado com frequência

Muitas aplicações que armazenam dados no Amazon S3 fornecem um “conjunto de trabalhos” de dados que são solicitados várias vezes pelos usuários. Se uma carga de trabalho estiver enviando solicitações GET repetidas para um conjunto comum de objetos, você poderá usar um cache, como o Amazon CloudFront, o Amazon ElastiCache ou o AWS Elemental MediaStore para otimizar a performance. A adoção bem-sucedida do cache pode resultar em baixar latência e altas taxas de transferência de dados. As aplicações que usam o armazenamento em cache também enviam menos solicitações diretas ao Amazon S3, o que também pode ajudar a reduzir os custos de solicitações.

O Amazon CloudFront é uma rede de entrega de conteúdo (CDN) rápida que armazena os dados do Amazon S3 em cache com transparência em um grande conjunto de pontos de presença (PoPs) distribuídos geograficamente. Quando os objetos podem ser acessados em várias regiões ou pela Internet, o CloudFront permite que os dados sejam armazenados em cache perto dos usuários que estão acessando os objetos. Isso pode resultar na entrega de alta performance de conteúdo popular do Amazon S3. Para obter mais informações sobre o CloudFront, consulte o Guia do desenvolvedor do Amazon CloudFront.

O Amazon ElastiCache é um cache de memória gerenciado. Com o ElastiCache, é possível provisionar instâncias do Amazon EC2 que armazenam objetos em cache na memória. Esse armazenamento em cache resulta em pedidos de redução de magnitude da latência de GET e aumentos significativos no throughput de download. Para usar o ElastiCache, modifique a lógica da aplicação para preencher o cache com objetos dinâmicos e verifique se esses objetos estão presentes no cache antes de solicitá-los do Amazon S3. Para obter exemplos de como usar o ElastiCache para melhorar a performance de GET do Amazon S3, consulte a publicação do blog Turbocharge Amazon S3 with Amazon ElastiCache for Redis.

O AWS Elemental MediaStore é um sistema de armazenamento em cache e de distribuição de conteúdo criado especificamente para fluxos de trabalho de vídeo e entrega de mídia do Amazon S3. O MediaStore fornece APIs de armazenamento completas especificamente para vídeo e é recomendado para workloads de vídeo sensíveis à performance. Para obter informações sobre o MediaStore, consulte o Guia do usuário do AWS Elemental MediaStore.

Tempo limite e repetição para aplicativos sensíveis à latência

Há determinadas situações em que uma aplicação recebe uma resposta do Amazon S3 indicando que uma nova tentativa é necessária. O Amazon S3 mapeia nomes de bucket e de objetos para os dados do objeto associados a eles. Se uma aplicação gerar altas taxas de solicitação (normalmente taxas constantes de mais de 5.000 solicitações por segundo para um pequeno número de objetos), ela poderá receber respostas HTTP 503 de lentidão. Se esses erros ocorrerem, cada SDK da AWS implementará uma lógica de repetição automática usando o recuo exponencial. Se você não estiver usando um SDK da AWS, implemente a lógica de repetição ao receber o erro HTTP 503. Para obter informações sobre técnicas de recuo, consulte Repetições de erro e recuo exponencial na AWS na Referência geral da Amazon Web Services.

O Amazon S3 é dimensionado automaticamente em resposta a novas taxas constantes de solicitação, otimizando a performance dinamicamente. Enquanto o Amazon S3 estiver sendo otimizado internamente para uma nova taxa de solicitação, você receberá respostas de solicitação HTTP 503 temporariamente até a otimização terminar. Depois que o Amazon S3 otimiza a performance internamente para a nova taxa de solicitação, todas as solicitações serão executadas de forma geral sem repetições.

Para aplicações sensíveis à latência, o Amazon S3 recomenda rastrear e repetir agressivamente as operações mais lentas. Ao repetir uma solicitação, recomendamos usar uma nova conexão ao Amazon S3 e executar uma nova pesquisa de DNS.

Ao fazer solicitações de tamanhos variavelmente grandes (por exemplo, mais de 128 MB), recomendamos rastrear o throughput atingido e repetir os 5% mais lentos das solicitações. Ao fazer solicitações menores (por exemplo, menos de 512 KB), onde latências medianas geralmente estão na faixa de dezenas de milissegundos, é recomendado repetir uma operação GET ou PUT depois de 2 segundos. Se outras repetições forem necessárias, é recomendado recuar. Por exemplo, recomendamos emitir uma repetição depois de 2 segundos e uma segunda repetição depois de mais 4 segundos.

Se a aplicação fizer solicitações de tamanho fixo para o Amazon S3, espere tempos de resposta mais consistentes para cada uma dessas solicitações. Nesse caso, uma estratégia simples é identificar o 1% mais lento de solicitações e repetí-las. Uma única repetição consegue reduzir a latência.

Se estiver usando o AWS Key Management Service (AWS KMS) para criptografia no lado do servidor, consulte Cotas no Guia do desenvolvedor do AWS Key Management Service para obter informações sobre as taxas de solicitações compatíveis com seu caso de uso.

Dimensionamento horizontal e paralelização de solicitações para alto throughput

O Amazon S3 é um sistema distribuído muito grande. Para ajudar a aproveitar essa escala, recomendamos dimensionar horizontalmente as solicitações paralelas para os endpoints do serviço Amazon S3. Além de distribuir as solicitações no Amazon S3, esse tipo de abordagem de dimensionamento ajuda a distribuir a carga em vários caminhos na rede.

Para obter altas taxas de transferência, o Amazon S3 recomenda usar aplicações com várias conexões para executar solicitações GET ou PUT paralelas de dados. Por exemplo, isso é compatível com o Amazon S3 Transfer Manager no AWS SDK para Java, e a maioria dos outros AWS SDKs fornece construções semelhantes. Para alguns aplicativos, você pode atingir conexões paralelas iniciando várias solicitações ao mesmo tempo em diferentes threads de aplicativo ou em diferentes instâncias de aplicativo. A melhor abordagem depende do aplicativo e da estrutura dos objetos acessados.

Você pode usar os AWS SDKs para emitir solicitações GET e PUT diretamente em vez de empregar o gerenciamento de transferências no AWS SDK. Essa abordagem permite ajustar a carga de trabalho mais diretamente, além de ainda aproveitar o suporte do SDK para repetições e o processamento das eventuais respostas HTTP 503. Como regra geral, ao fazer download de grandes objetos em uma região do Amazon S3 para o Amazon EC2, sugerimos fazer solicitações

simultâneas em escalas de bytes de um objeto na granularidade de 8 a 16 MB. Faça uma solicitação simultânea para cada 85 a 90 MB/s da taxa de transferência de rede desejada. Para saturar uma placa de interface de rede (NIC) de 10 Gb/s, você pode usar cerca de 15 solicitações simultâneas em conexões separadas. É possível dimensionar as solicitações simultâneas em mais conexões para saturar NICs mais rápidas, como NICs de 25 Gb/s ou 100 Gb/s.

A avaliação do desempenho é importante ao ajustar o número de solicitações a serem emitidas ao mesmo tempo. Recomendamos começar com uma única solicitação de cada vez. Meça a largura de banda de rede atingida e o uso de outros recursos que o aplicativo usa no processamento dos dados. Você pode identificar o recurso de gargalo (isto é, o recurso com maior utilização) e, assim, o número de solicitações que provavelmente serão úteis. Por exemplo, se processar uma solicitação por vez usa 25% da CPU, é recomendado acomodar até quatro solicitações simultâneas.

A medição é essencial e vale a pena confirmar o uso do recurso conforme a taxa de solicitação aumenta.

Se a aplicação emitir solicitações diretamente para o Amazon S3 usando a API REST, recomendamos usar um grupo de conexões HTTP e reutilizar cada conexão para uma série de solicitações. Evitar a configuração de conexão por solicitação elimina a necessidade de realizar handshakes Secure Sockets Layer (SSL) e TCP de inicialização lenta em cada solicitação. Para obter informações sobre como usar a API REST, consulte a Apresentação da API REST do Amazon S3.

Finalmente, vale prestar atenção ao DNS e verificar novamente se as solicitações estão sendo distribuídas em um grande grupo de endereços IP do Amazon S3. As consultas de DNS para o Amazon S3 percorrem uma grande lista de endpoints IP. No entanto, o armazenamento em cache de solucionadores ou código do aplicativo que reutiliza um único endereço IP não aproveita a diversidade de endereços e o balanceamento de carga associados. Ferramentas de utilitário de rede, como a ferramenta de linha de comando `netstat`, podem mostrar os endereços IP usados para comunicação com o Amazon S3, e nós fornecemos diretrizes para as configurações de DNS que devem ser usadas. Para obter mais informações sobre essas diretrizes, consulte Roteamento de solicitações.

Usar o Amazon S3 Transfer Acceleration para acelerar transferências de dados geograficamente dispersas

O Amazon S3 Transfer Acceleration é útil para minimizar ou eliminar a latência causada pela distância geográfica entre clientes distribuídos globalmente e uma aplicação regional que usa

o Amazon S3. O Transfer Acceleration usa os pontos de presença distribuídos globalmente no CloudFront para transporte de dados. A rede de presença da AWS tem pontos de presença em mais de 50 locais. Atualmente, ela é usada para distribuir conteúdo por meio do CloudFront e fornecer respostas rápidas para consultas de DNS feitas para o Amazon Route 53.

A rede de borda também ajuda a acelerar transferências de dados enviadas e recebidas do Amazon S3. Ela é ideal para aplicativos que transferem dados em ou entre continentes, possuem uma rápida conexão com a Internet, usam objetos grandes ou possuem muito conteúdo para upload. Conforme os dados chegam em um ponto de presença, eles são roteados para o Amazon S3 por um caminho de rede otimizado. Em geral, quanto mais distante você está de uma região do Amazon S3, maior a melhoria de velocidade que pode esperar do uso do Transfer Acceleration.

Você pode configurar o Transfer Acceleration em buckets novos ou existentes. Use um endpoint separado do Amazon S3 Transfer Acceleration para usar os pontos de presença da AWS. A melhor maneira de testar se o Transfer Acceleration ajuda a performance da solicitação do cliente é usar a ferramenta de comparação de velocidade do Amazon S3 Transfer Acceleration. As configurações e condições de rede variam periodicamente e de um lugar para outro. Portanto, você só é cobrado por transferências em que o Amazon S3 Transfer Acceleration pode melhorar a performance do upload. Para obter informações sobre como usar o Transfer Acceleration com diferentes AWS SDKs, consulte Exemplos do Amazon S3 Transfer Acceleration.

Colaboradores

Os colaboradores desse documento incluem:

- Mai-Lan Tomsen Bukovec, vice-presidente do Amazon S3
- Andy Warfield, engenheiro-chefe sênior do Amazon S3
- Tim Harris, engenheiro-chefe do Amazon S3

Revisões do documento

Para ser notificado sobre atualizações deste whitepaper, inscreva-se no RSS feed.

update-history-change	update-history-description	update-history-date
Atualizado	Revisado quanto à precisão técnica	10 de março de 2021
Publicação inicial	Publicação inicial	1 de junho de 2019

Avisos

Os clientes são responsáveis por fazer sua própria avaliação independente das informações neste documento. Este documento é: (a) fornecido apenas para fins informativos, (b) representa as ofertas e práticas de produtos atuais da AWS, que estão sujeitas a alterações sem aviso prévio e (c) não cria nenhum compromisso ou garantia da AWS e suas afiliadas, fornecedores ou licenciadores. Os produtos ou serviços da AWS são fornecidos no “estado em que se encontram”, sem garantias, declarações ou condições de qualquer tipo, explícitas ou implícitas. As responsabilidades e obrigações da AWS com seus clientes são regidas por contratos da AWS, e este documento não modifica nem faz parte de nenhum contrato entre a AWS e seus clientes.