Whitepaper AWS

Modelli di progettazione delle best practice: ottimizzazione delle prestazioni di Amazon S3

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

Modelli di progettazione delle best practice: ottimizzazione delle prestazioni di Amazon S3: Whitepaper AWS

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

I marchi e il trade dress di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in qualsiasi modo che possa causare confusione tra i clienti o in qualsiasi modo che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà dei rispettivi proprietari, che possono o meno essere affiliati, collegati o sponsorizzati da Amazon.

Table of Contents

Sintesi 1
Riassunto 1
Introduzione 2
Linee guida sulle prestazioni per Amazon S3 4
Misura le prestazioni 4
Scala le connessioni di storage orizzontalmente 4
Utilizza i fetches Byte-Range 5
Riprova le richieste per le applicazioni sensibili alla latenza 5
Combina Amazon S3 (storage) e Amazon EC2 (calcolo) nella stessa regione AWS 5
Utilizza Amazon S3 Transfer Acceleration per ridurre al minimo la latenza causata dalla distanza 5
Scegli la versione più recente degli SDK AWS 6
Schemi di progettazione per le prestazioni di Amazon S3 7
Utilizzo del caching per i contenuti ad accesso frequente 7
Timeout e tentativi per applicazioni sensibili alla latenza 8
Dimensionamento orizzontale e parallelizzazione delle richieste per throughput elevato 9
Utilizzo di Amazon S3 Transfer Acceleration per accelerare trasferimenti di dati in zone geografiche lontane 10
Collaboratori 12
Revisioni del documento 13
Avvisi 14

Modelli di progettazione delle best practice: ottimizzazione delle prestazioni di Amazon S3

Data di pubblicazione iniziale: giugno 2019 (Revisioni del documento)

Riassunto

Per sviluppare applicazioni che caricano e recuperano oggetti da Amazon S3, segui le linee guida AWS sulle best practice per ottimizzare le prestazioni. AWS offre anche modelli di progettazione delle prestazioni più dettagliati.

Introduzione

Le applicazioni possono facilmente raggiungere migliaia di transazioni al secondo come prestazioni delle richieste durante il caricamento e il recupero di risorse di storage da Amazon S3. Amazon S3 si ridimensiona automaticamente fino a tassi di richiesta elevati. Ad esempio, l'applicazione può raggiungere almeno 3.500`PUT/COPY/POST/DELETE` e 5.500`GET/HEAD` richieste al secondo per prefisso in un bucket. Non esistono limiti al numero di prefissi in un bucket. È possibile aumentare le proprie performance in lettura o scrittura parallelizzando le scritture. Ad esempio, se si creano 10 prefissi in un bucket Amazon S3 per parallelizzare le letture, è possibile scalare le prestazioni di lettura a 55.000 richieste di lettura al secondo.

Alcune applicazioni di data lake in Amazon S3 scansionano diversi milioni o miliardi di oggetti per query eseguite su diversi petabyte di dati. Queste applicazioni di data lake raggiungono tassi di trasferimento per singola istanza che massimizzano l'utilizzo dell'interfaccia di rete per l'istanza Amazon EC2, arrivando fino a 100 Gb/s su una singola istanza. Queste applicazioni poi aggregano throughput su più istanze per ottenere diversi terabit al secondo.

Altre applicazioni sono sensibili alla latenza, come le applicazioni di messaggistica sui social media. Tali applicazioni possono raggiungere latenze coerenti per oggetti di piccole dimensioni (e latenze al primo byte in uscita per oggetti più grandi) di circa 100-200 millisecondi.

Altri servizi AWS possono accelerare le prestazioni per diverse architetture dell'applicazione. Ad esempio, se desideri velocità di trasferimento più elevate su una singola connessione HTTP o latenze del valore di millisecondi a una cifra, utilizza Amazon CloudFront o Amazon ElastiCache per il caching con Amazon S3.

Inoltre, se desideri trasferimenti dei dati veloci su lunghe distanze tra un client e un bucket S3, usa Amazon S3 Transfer Acceleration. Transfer Acceleration utilizza le edge location distribuite a livello globale in CloudFront per accelerare il trasferimento dei dati su distanze geografiche.

Se il carico di lavoro di Amazon S3 utilizza la crittografia lato server con AWS Key Management Service (SSE-KMS), consulta la sezione AWS KMSLimiti nella AWS Key Management ServiceGuida per gli sviluppatori per informazioni sui tassi di richiesta supportati per il tuo caso d'uso.

Gli argomenti seguenti presentano le linee guida e i modelli di progettazione per le best practice per ottimizzare le prestazioni per le applicazioni che utilizzano Amazon S3.

Queste linee guida sostituiscono eventuali linee guida precedenti sull'ottimizzazione delle prestazioni per Amazon S3. Ad esempio, le linee guida per le prestazioni di Amazon S3 consigliavano in

precedenza di utilizzare una denominazione casuale dei prefissi con caratteri con hash per ottimizzare le prestazioni per recuperi frequenti di dati. Ora non è più necessario scegliere in maniera casuale la denominazione dei prefissi per motivi di performance ed è possibile utilizzare denominazioni sequenziali basate sulla data. Consulta le linee guida sulle prestazioni e i modelli di progettazione delle prestazioni per le informazioni più aggiornate sull'ottimizzazione delle prestazioni per Amazon S3.

Linee guida sulle prestazioni per Amazon S3

Per ottenere prestazioni ottimali per le applicazioni su Amazon S3, AWS consiglia di adottare le linee guida seguenti.

Argomenti

- Misura le prestazioni
- Scala le connessioni di storage orizzontalmente
- Utilizza i fetches Byte-Range
- Riprova le richieste per le applicazioni sensibili alla latenza
- Combina Amazon S3 (storage) e Amazon EC2 (calcolo) nella stessa regione AWS
- Utilizza Amazon S3 Transfer Acceleration per ridurre al minimo la latenza causata dalla distanza
- Scegli la versione più recente degli SDK AWS

Misura le prestazioni

Quando si ottimizzano le prestazioni, devi considerare i requisiti di throughput di rete, CPU e DRAM (Dynamic Random Access Memory). A seconda della combinazione di esigenze per queste risorse diverse, può essere utile valutare più tipi di istanza Amazon EC2. Per informazioni sui tipi di istanza disponibili, consulta Tipi di istanza nella Guida per l'utente di Amazon EC2 per istanze Linux.

Durante la misurazione delle prestazioni, è utile anche verificare i tempi, la latenza e la velocità del trasferimento dei dati utilizzando strumenti di analisi HTTP.

Scala le connessioni di storage orizzontalmente

Distribuire le richieste su più connessioni è uno schema di progettazione comune per scalare orizzontalmente le prestazioni. Se devi creare applicazioni ad alte prestazioni, pensa ad Amazon S3 come un sistema distribuito di dimensioni molto grandi, non un singolo endpoint di rete come un server di storage tradizionale. Puoi ottenere prestazioni ottimali inviando più richieste simultanee ad Amazon S3. Distribuisci queste richieste su connessioni separate per massimizzare la larghezza di banda accessibile da Amazon S3. Amazon S3 non impone limiti al numero di connessioni effettuate al bucket.

Utilizza i fetches Byte-Range

Utilizzando l'intestazione HTTP Range in una richiesta GET Object, puoi recuperare un intervallo di byte da un oggetto, trasferendo solo la parte specificata. Puoi utilizzare connessioni simultanee ad Amazon S3 per recuperare diversi intervalli di byte all'interno dello stesso oggetto. Questa operazione ti permette di ottenere un throughput aggregato superiore rispetto a una singola richiesta whole-object. Recuperare range minori da oggetti più grandi permette inoltre alla tua applicazione di migliorare i tempi di ripetizione quando le richieste sono interrotte. Per ulteriori informazioni, consulta Recupero di oggetti.

Le dimensioni tipiche per le richieste byte-range sono di 8 o 16 MB. Se gli oggetti sono oggetti PUT che utilizzano un caricamento in più parti, è buona pratica trasformarli in oggetti GET nelle stesse dimensioni della parte (o almeno allineati ai limiti della parte) per ottenere le prestazioni migliori. Le richieste GET possono rivolgersi direttamente alle singole parti; ad esempio, `GET ?partNumber=N`

Riprova le richieste per le applicazioni sensibili alla latenza

I timeout e i tentativi aggressivi aiutano a mantenere la latenza uniforme. Poiché Amazon S3 opera su vasta scala, se la prima richiesta è lenta, un nuovo tentativo di richiesta adotterà un percorso diverso e riuscirà rapidamente. Gli SDK AWS hanno valori di timeout e tentativo configurabili che puoi regolare sulle tolleranze della tua applicazione.

Combina Amazon S3 (storage) e Amazon EC2 (calcolo) nella stessa regione AWS

Sebbene i nomi dei bucket S3 siano univoci a livello globale, ogni bucket viene archiviato in una regione che sceglierai durante la creazione del bucket stesso. Per ottimizzare le prestazioni, consigliamo di accedere al bucket dalle istanze Amazon EC2 nella stessa regione AWS, se possibile. Questa operazione permette di ridurre la latenza e i costi di trasferimento dei dati.

Per ulteriori informazioni sui costi dl trasferimento dei dati, consulta Prezzi di Amazon S3.

Utilizza Amazon S3 Transfer Acceleration per ridurre al minimo la latenza causata dalla distanza

Amazon S3 Transfer Acceleration gestisce trasferimenti rapidi, semplici e sicuri di file su lunghe distanze geografiche tra il client e un bucket S3. Transfer Acceleration sfrutta le edge location

distribuite a livello globale in Amazon CloudFront. Quando arrivano in una edge location, i dati vengono instradati ad Amazon S3 attraverso un percorso di rete ottimizzato. Transfer Acceleration è ideale per il trasferimento regolare di gigabyte e terabyte di dati sui diversi continenti. È inoltre utile per i clienti che effettuano il caricamento in un bucket centralizzato da tutto il mondo.

Puoi utilizzare lo strumento Speed Comparison di Amazon S3 Transfer Acceleration per confrontare le velocità di caricamento accelerate e non accelerate tra regioni Amazon S3. Questo strumento utilizza caricamenti in più parti per trasferire un file dal browser in uso a diverse regioni Amazon S3 con e senza l'utilizzo di Amazon S3 Transfer Acceleration.

Scegli la versione più recente degli SDK AWS

Gli SDK AWS offrono supporto integrato per molte delle linee guida consigliate per l'ottimizzazione delle prestazioni di Amazon S3. Gli SDK offrono un'API più semplice per utilizzare al meglio Amazon S3 internamente a un'applicazione e vengono aggiornati regolarmente in modo da seguire le best practice più recenti. Ad esempio, gli SDK includono la logica per riprovare automaticamente le richieste sugli errori HTTP 503 e investono in codice per rispondere e adattarsi a connessioni lente.

Gli SDK forniscono anche lo strumento Transfer Manager, che automatizza connessioni con dimensionamento orizzontale in modo da raggiungere migliaia di richieste al secondo, usando richieste di intervalli di byte laddove appropriato. È importante utilizzare l'ultima versione degli SDK AWS per ottenere le funzionalità di ottimizzazione delle prestazioni più recenti.

Puoi inoltre ottimizzare le prestazioni quando utilizzi le richieste dell'API REST HTTP. Quando utilizzi l'API REST, devi seguire le stesse best practice che fanno parte degli SDK. Consenti i timeout e i tentativi sulle richieste lente e le connessioni multiple per permettere il recupero dei dati degli oggetti in parallelo. Per informazioni sull'utilizzo dell'API REST, consulta la Documentazione di riferimento delle API di Amazon Simple Storage Service.

Schemi di progettazione per le prestazioni di Amazon S3

Nel progettare applicazioni per caricare e recuperare l'archiviazione da Amazon S3, utilizza i nostri schemi di progettazione delle best practice per ottenere prestazioni ottimali per l'applicazione. Offriamo anche linee guida sulle prestazioni da tenere in considerazione quando si pianifica l'architettura dell'applicazione.

Per ottimizzare le prestazioni, puoi utilizzare i seguenti schemi di progettazione.

Argomenti

- Utilizzo del caching per i contenuti ad accesso frequente
- Timeout e tentativi per applicazioni sensibili alla latenza
- Dimensionamento orizzontale e parallelizzazione delle richieste per throughput elevato
- Utilizzo di Amazon S3 Transfer Acceleration per accelerare trasferimenti di dati in zone geografiche lontane

Utilizzo del caching per i contenuti ad accesso frequente

Molte applicazioni che archiviano dati in Amazon S3 fungono da "working set" dei dati richiesti ripetutamente dagli utenti. Se un carico di lavoro invia richieste GET ripetute di un set comune di oggetti, puoi utilizzare una cache come Amazon CloudFront, Amazon ElastiCache, o AWS Elemental MediaStore per ottimizzare le prestazioni. L'adozione corretta di una cache può portare a una bassa latenza e a tassi di trasferimento dei dati più alti. Le applicazioni che utilizzano il caching inviano anche meno richieste dirette ad Amazon S3, riducendo i costi delle richieste.

Amazon CloudFront è una rete di distribuzione di contenuti veloce che esegue il caching trasparente dei dati da Amazon S3 in un set di punti di presenza distribuiti in varie regioni di grandi dimensioni. Quando gli oggetti sono accessibili da più regioni o su Internet, CloudFront permette il caching dei dati vicino agli utenti che accedono a tali oggetti. In questo modo, è possibile distribuire contenuti Amazon S3 comuni con prestazioni elevate. Per ulteriori informazioni su CloudFront, consulta la Guida per gli sviluppatori di Amazon CloudFront.

Amazon ElastiCache è una cache in memoria gestita. Con ElastiCache, puoi effettuare il provisioning di istanze Amazon EC2 che eseguono il caching di oggetti in memoria. Il caching porta alla riduzione di grandezza della latenza GET e ad aumenti sostanziali nel throughput del download. Per utilizzare

ElastiCache, devi modificare la logica dell'applicazione in modo da popolare la cache con oggetti ad accesso frequente e controllare la presenza di tali oggetti nella cache prima di richiederli da Amazon S3. Per alcuni esempi di utilizzo di ElastiCache per migliorare le prestazioni delle richieste GET in Amazon S3, consulta il post di blog relativo al potenziamento di Amazon S3 con Amazon ElastiCache for Redis.

AWS Elemental MediaStore è un sistema di caching e distribuzione di contenuti progettato appositamente per flussi di lavoro video e distribuzione di contenuti multimediali da Amazon S3. MediaStore fornisce API di storage end-to-end specifiche per i contenuti video ed è consigliato per carichi di lavoro video sensibili alle prestazioni. Per informazioni su MediaStore, consulta la Guida per l'utente di AWS Elemental MediaStore.

Timeout e tentativi per applicazioni sensibili alla latenza

In alcune situazioni un'applicazione riceve una risposta da Amazon S3 che indica che è necessario un nuovo tentativo. Amazon S3 mappa i nomi dei bucket e degli oggetti ai dati degli oggetti a essi associati. Se un'applicazione genera alti tassi di richiesta (in genere tassi sostenuti di oltre 5.000 richieste al secondo per un piccolo numero di oggetti) potrebbe ricevere risposte di rallentamento HTTP 503. Se si verifica questo errore, ogni SDK AWS implementa la logica di tentativo automatica utilizzando il backoff esponenziale. Se non stai utilizzando un SDK AWS, devi implementare la logica di tentativo quando ricevi un errore HTTP 503. Per ulteriori informazioni sulle tecniche di back-off, consulta Ripetizione dei tentativi in caso di errore e backoff esponenziale in AWS in Riferimenti generali Amazon Web Services.

Amazon S3 si ridimensiona automaticamente in risposta a nuovi tassi di richiesta prolungati, ottimizzando dinamicamente le prestazioni. Mentre Amazon S3 ottimizza internamente per sostenere un nuovo tasso di richieste, riceverai temporaneamente risposte di richiesta HTTP 503 fino al completamento dell'ottimizzazione. Dopo che Amazon S3 ha ottimizzato internamente le prestazioni per il nuovo tasso di richiesta, tutte le richieste vengono in genere gestite senza nuovi tentativi.

Per le applicazioni sensibili alla latenza, Amazon S3 consiglia di monitorare e ritentare in modo aggressivo le operazioni più lente. Nel ritentare una richiesta, consigliamo di utilizzare una nuova connessione ad Amazon S3 e di eseguire una nuova ricerca DNS.

Quando effettui richieste di dimensioni grandi e variabili (ad esempio, oltre 128 MB), consigliamo di tracciare il throughput raggiunto e di ritentare il 5 percento più lento delle richieste. Quando effettui richieste più piccole (ad esempio, meno di 512 KB) dove le latenze medie sono spesso dell'ordine di decine di millisecondi, una buona linea guida è ritentare un'operazione GET o PUT

dopo 2 secondi. Se sono necessari tentativi aggiuntivi, la best practice è di effettuare il backoff. Ad esempio, consigliamo di emettere un tentativo dopo 2 secondi e un secondo tentativo dopo 4 secondi aggiuntivi.

Se l'applicazione effettua richieste a dimensione fissa ad Amazon S3, il tempo di risposta per ogni richiesta sarà più costante. In questo caso, una strategia semplice è identificare l'1 percento più lento delle richieste e ritentarle. Anche un singolo tentativo è efficace nella riduzione della latenza.

Se utilizzi AWS Key Management Service (AWS KMS) per la crittografia lato server, consulta Limiti nella AWS Key Management ServiceGuida per gli sviluppatori per informazioni sui tassi di richiesta supportati per il tuo caso d'uso.

Dimensionamento orizzontale e parallelizzazione delle richieste per throughput elevato

Amazon S3 è un sistema distribuito di dimensioni molto grandi. Per sfruttarne a pieno la capacità di dimensionamento, consigliamo di ridimensionare orizzontalmente le richieste parallele agli endpoint del servizio Amazon S3. Oltre a distribuire le richieste in Amazon S3, questo tipo di approccio al dimensionamento permette di distribuire il carico su più percorsi nella rete.

Per i trasferimenti a throughput elevato, Amazon S3 consiglia di utilizzare applicazioni che usano più connessioni a dati GET o PUT in parallelo. Ad esempio, questo approccio è supportato da Transfer Manager in Amazon S3 nell'SDK Java AWS e la maggior parte degli SDK AWS offre costrutti simili. Per alcune applicazioni, puoi raggiungere connessioni parallele lanciando simultaneamente richieste multiple in diversi thread dell'applicazione o in diverse istanze dell'applicazione. Il miglior approccio da adottare dipende dall'applicazione e dalla struttura degli oggetti a cui accedi.

Puoi utilizzare gli SDK AWS per emettere direttamente richieste GET e PUT anziché impiegare la gestione dei trasferimenti negli SDK AWS. Questo approccio ti permette di ottimizzare il carico di lavoro in modo più diretto senza rinunciare al supporto degli SDK per i tentativi e la gestione delle risposte HTTP 503 che potrebbero verificarsi. Come regola generale, quando scarichi oggetti di grandi dimensioni all'interno di una regione da Amazon S3 ad Amazon EC2, consigliamo di effettuare richieste simultanee di intervalli di byte di un oggetto in base a una granularità di 8–16 MB. Effettua una richiesta simultanea per ogni 85–90 MB/s di throughput di rete desiderato. Per saturare una network interface card (NIC) da 10 Gb/s, devi utilizzare circa 15 richieste simultanee su connessioni separate. Puoi scalare le richieste simultanee su più connessioni per saturare più rapidamente le NIC, come quelle da 25 Gb/s o 100 Gb/s.

La misurazione delle prestazioni è importante quando ottimizzi il numero di richieste da emettere simultaneamente. Consigliamo di iniziare con un richiesta alla volta. Misura la larghezza di banda della rete raggiunta e l'uso delle altre risorse che la tua applicazione utilizza nell'elaborazione dei dati. Puoi quindi identificare la risorsa con un collo di bottiglia (ossia, la risorsa con l'utilizzo più elevato) e di conseguenza il numero di richieste che possono essere utili. Ad esempio, se elaborare una richiesta alla volta porta a un utilizzo della CPU del 25 percento, questo dato suggerisce che possono essere emesse fino a quattro richieste simultanee.

La misurazione è essenziale ed è utile per confermare l'utilizzo della risorsa quando il tasso di richiesta aumenta.

Se l'applicazione invia richieste direttamente ad Amazon S3 utilizzando l'API REST, ti consigliamo di utilizzare un pool di connessioni HTTP e di riutilizzare ogni connessione per una serie di richieste. Evitare la configurazione della connessione per richiesta elimina la necessità di eseguire handshake slow-start su TCP e Secure Sockets Layer (SSL) su ogni richiesta. Per informazioni sull'utilizzo dell'API REST, consulta l'Introduzione all'API REST di Amazon S3.

Infine, è utile considerare con attenzione DNS e verificare accuratamente che le richieste vengano distribuite su un ampio pool di indirizzi IP di Amazon S3. Le query DNS per Amazon S3 passano per un elenco di grandi dimensioni di endpoint IP. Ma effettuare il caching dei resolver o del codice dell'applicazione che riutilizza un singolo indirizzo IP non trae vantaggio dalla diversità degli indirizzi e dal bilanciamento del carico che ne deriva. Utilità di rete come lo strumento a riga di comando `netstat` possono mostrare gli indirizzi IP utilizzati per la comunicazione con Amazon S3 e forniamo linee guida per le configurazioni DNS da utilizzare. Per ulteriori informazioni su queste linee guida, consulta Routing delle richieste.

Utilizzo di Amazon S3 Transfer Acceleration per accelerare trasferimenti di dati in zone geografiche lontane

Amazon S3 Transfer Acceleration è efficace nel ridurre o eliminare la latenza causata dalla distanza geografica tra client lontani a livello globale e un'applicazione locale che utilizza Amazon S3. Transfer Acceleration utilizza le edge location distribuite a livello globale in CloudFront per il trasferimento dei dati. La rete edge AWS dispone di punti di presenza in oltre 50 località. Oggi viene utilizzata per distribuire contenuti tramite CloudFront e per offrire risposte rapide a query DNS eseguite su Amazon Route 53.

La rete edge permette anche di accelerare i trasferimenti dei dati da e verso Amazon S3. È ideale per le applicazioni che trasferiscono i dati tra continenti, dispongono di connessioni a Internet veloci

e utilizzano oggetti di grandi dimensioni o hanno molti contenuti da caricare. Quando arrivano in una edge location, i dati vengono instradati ad Amazon S3 su un percorso di rete ottimizzato. In generale, più lontano ti trovi da una regione Amazon S3, maggiore è il miglioramento della velocità che otterrai utilizzando Transfer Acceleration.

Puoi configurare Transfer Acceleration su bucket nuovi o esistenti. Puoi utilizzare un endpoint Amazon S3 Transfer Acceleration separato per utilizzare le edge location AWS. Il modo migliore per verificare se Transfer Acceleration migliora le prestazioni delle richieste client consiste nell'utilizzare lo strumento Speed Comparison di Amazon S3 Transfer Acceleration. Le configurazioni e le condizioni della rete variano in base al momento e alla località. Vengono quindi addebitati solo i trasferimenti in cui Amazon S3 Transfer Acceleration può potenzialmente migliorare le prestazioni di caricamento. Per informazioni sull'utilizzo di Transfer Acceleration con diversi SDK AWS, consulta Esempi di Amazon S3 Transfer Acceleration.

Collaboratori

I collaboratori di questo documento includono:

- Mai-Lan Tomsen Bukovec, vicepresidente (VP) di Amazon S3
- Andy Warfield, ingegnere principale senior, Amazon S3
- Tim Harris, ingegnere principale, Amazon S3

Revisioni del documento

Per ricevere una notifica sugli aggiornamenti di questo whitepaper, iscriviti al feed RSS.

update-history-change	update-history-description	update-history-date
Aggiornato	Rivisto per l'accuratezza tecnica	10 marzo 2021
Pubblicazione iniziale	Pubblicazione iniziale	1 giugno 2019

Avvisi

I clienti sono responsabili della propria valutazione autonoma delle informazioni contenute in questo documento. Questo documento: (a) è solo a scopo informativo, (b) mostra le offerte e le pratiche attuali dei prodotti AWS soggette a modifiche senza preavviso, e (c) non crea alcun impegno o garanzia da parte di AWS e dei suoi affiliati, fornitori o licenziatari. I prodotti o servizi AWS sono forniti "così come sono" senza garanzie, dichiarazioni o condizioni di alcun tipo, sia esplicite che implicite. Le responsabilità e gli obblighi di AWS verso i propri clienti sono disciplinati dagli accordi AWS e il presente documento non fa parte né modifica alcun accordo tra AWS e i suoi clienti.

© 2020, Amazon Web Services, Inc. o sue affiliate. Tutti i diritti riservati.