



ElastiCache per la guida dell'utente di Memcached

Amazon ElastiCache



Versione API 2015-02-02

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

Amazon ElastiCache: ElastiCache per la guida dell'utente di Memcached

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

I marchi e l'immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà delle rispettive aziende, che possono o meno essere associate, collegate o sponsorizzate da Amazon.

Table of Contents

Che cos'è ElastiCache per Memcached?	1
Cache serverless	1
Cluster progettati autonomamente	2
Come funziona	2
Motori di cache e caching	2
Casi d'uso	8
Datastore in memoria	8
Testimonianze dei clienti su ElastiCache	10
Risorse di Memcached ElastiCache	10
Strumenti per la gestione dell'implementazione	12
Utilizzo di AWS Management Console	12
Utilizzo di AWS CLI	12
Utilizzo dell'SDK AWS	12
Utilizzo dell'API ElastiCache	12
Consulta anche	13
Scelta tra le opzioni di implementazione	13
Confronto tra cache progettate autonomamente Memcached e Redis	14
Nozioni di base su ElastiCache for MemCached	20
Configurazione	20
Registrati per un Account AWS	20
Creazione di un utente amministratore	21
Concessione dell'accesso programmatico	22
Impostazione delle autorizzazioni	24
Configurazione di EC2	25
Assegnazione dell'accesso di rete	25
Crea una cache.	26
Creazione di una cache serverless	26
Lettura e scrittura di dati	27
(Facoltativo) Pulizia	32
Fasi successive	33
Tutorial: Configurazione di una funzione Lambda per l'accesso ad Amazon ElastiCache in un Amazon VPC	34
Fase 1: creazione di una cache ElastiCache	34
Passaggio 2: creazione di una funzione Lambda	36

Fase 3: esecuzione del test della funzione Lambda	41
Tutorial e video	42
Video	43
Scelta di regioni e zone di disponibilità	49
Considerazioni sulle zone di disponibilità	50
Regioni ed endpoint supportati	52
Ubicazione dei nodi	58
Utilizzo di Local Zones	58
Abilitazione di una zona locale	59
Utilizzo Outposts	59
Utilizzo degli Outposts con la console Memcached	60
Utilizzo degli Outposts con il CLI AWS	61
Progettazione del proprio cluster ElastiCache	63
ElastiCache per componenti e funzionalità Memcached	64
Nodi	64
Cluster	65
AWS Regioni e zone di disponibilità	66
Endpoints	67
Gruppi di parametri	67
Sicurezza	68
Gruppi di sottoreti	68
Eventi	68
Gestione dei cluster	69
Scelta del tipo di rete	71
Individuazione automatica	73
Preparazione di un cluster	116
Creazione di un cluster	123
Visualizzazione dei dettagli di un cluster	126
Modifica di un cluster	131
Riavvio di un cluster	135
Aggiunta di nodi a un cluster	137
Rimozione di nodi da un cluster	143
Annullamento delle operazioni in attesa di aggiunta e di eliminazione dei nodi	149
Eliminazione di un cluster	150
Accesso al cluster	152
Individuazione degli endpoint di connessione	158

Gestione di nodi	166
Visualizzazione dello stato ElastiCache del nodo	166
Connessione ai nodi	172
Tipi di nodi supportati	175
Sostituzione dei nodi	185
Nodi riservati	187
Migrazione dei nodi della generazione precedente	199
Lavorare con ElastiCache	201
Versioni e aggiornamenti del motore	201
Versioni di Memcached supportate	202
Versioni e aggiornamenti del motore	206
Come aggiornare la versione di un motore	207
Best practice e strategie di caching	208
Best practice con i client Memcached	208
Comandi Memcached supportati	216
Strategie di caching	217
Gestione del cluster progettato autonomamente	222
Gestione della manutenzione	223
Configurazione dei parametri di motore con i gruppi di parametri	225
Scalabilità ElastiCache per Memcached	269
Scalabilità ElastiCache per Memcached	269
Impostazione dei limiti di dimensionamento per gestire i costi	269
Prescalabilità con Serverless ElastiCache	269
Impostazione dei limiti di scalabilità utilizzando la console e AWS CLI	271
Scalabilità ElastiCache per cluster progettati autonomamente da Memcached	272
Assegnazione di tag delle risorse ElastiCache	276
Monitoraggio dei costi con i tag	283
Gestione dei tag con la AWS CLI	285
Gestione dei tag con l'API di ElastiCache	288
Amazon ElastiCache Well-Architected Lens	290
Pilastro dell'eccellenza operativa	291
Pilastro della sicurezza	300
Pilastro dell'affidabilità	306
Pilastro dell'efficienza delle prestazioni	312
Pilastro dell'ottimizzazione dei costi	323
Risoluzione dei problemi	329

Problemi di connessione	330
Errori del client Redis	330
Risoluzione dei problemi di latenza elevata in Serverless ElastiCache	331
Risoluzione dei problemi di limitazione in Serverless ElastiCache	332
Argomenti correlati	333
Ulteriori passaggi per la risoluzione dei problemi	334
Gruppi di sicurezza	334
Liste di controllo accessi (ACL) di rete	335
Tabelle di instradamento	336
Risoluzione DNS	337
Identificazione dei problemi con la diagnostica lato server	337
Convalida della connettività di rete	343
Limiti relativi alla rete	345
Utilizzo CPU	346
Connessioni terminate dal lato server	350
Risoluzione dei problemi lato client per istanze Amazon EC2	351
Dissezione del tempo necessario per completare una singola richiesta	352
Sicurezza	355
Protezione dei dati	356
Sicurezza dei dati in Amazon ElastiCache	356
Riservatezza del traffico Internet	366
Amazon VPC e sicurezza di ElastiCache	366
API Amazon ElastiCache ed endpoint VPC di interfaccia (AWS PrivateLink)	389
Sottoreti e gruppi di sottoreti	392
Identity and Access Management	401
Destinatari	401
Autenticazione con identità	402
Gestione dell'accesso con policy	406
Funzionamento di Amazon ElastiCache con IAM	408
Esempi di policy basate su identità	415
Risoluzione dei problemi	419
Controllo accessi	421
Panoramica sulla gestione degli accessi	422
Convalida della conformità	454
Ulteriori informazioni	456
Resilienza	456

Limitazione dell'impatto degli errori	456
Sicurezza dell'infrastruttura	458
Aggiornamenti di servizio	459
Gestione degli aggiornamenti del servizio	459
Registrazione di log e monitoraggio	466
Metriche ed eventi serverless	466
Metriche serverless	466
Eventi serverless	470
Metriche ed eventi progettati autonomamente	475
Metriche dei cluster progettati autonomamente	475
Eventi di cluster progettati autonomamente	476
Monitoraggio dell'utilizzo	483
Monitoraggio degli eventi di Amazon SNS	497
Registrazione delle chiamate all'API Amazon ElastiCache con AWS CloudTrail	513
Informazioni su Amazon ElastiCache in CloudTrail	514
Informazioni sulle voci dei file di log di Amazon ElastiCache	515
Quote	519
Riferimento	520
Utilizzo dell'API ElastiCache	520
Uso dell'API query	520
Librerie disponibili	524
Risoluzione dei problemi delle applicazioni	524
Impostazione dell'interfaccia AWS CLI ElastiCache	525
Prerequisiti	526
Ottenere gli strumenti a riga di comando	527
Configurazione degli strumenti	528
Fornire le credenziali per gli strumenti	529
Variabili di ambiente	530
Messaggi di errore	531
Notifiche	532
Notifiche generali ElastiCache	533
Notifiche ElastiCache per Memcached	533
.....	534
AWS Glossario	551
.....	dlii

Cos'è Amazon ElastiCache per Memcached?

Benvenuto nella Guida per l'utente di Amazon ElastiCache for Memcached. Amazon ElastiCache è un servizio Web che semplifica la configurazione, la gestione e la scalabilità di un archivio dati in memoria distribuito o di un ambiente di cache nel cloud. Fornisce una soluzione di caching scalabile ad alte prestazioni e a costi contenuti. Allo stesso tempo, aiuta a rimuovere la complessità associata all'implementazione e alla gestione di un ambiente di cache distribuita.

Puoi utilizzare Amazon ElastiCache in due formati. Puoi iniziare con una cache serverless o scegliere di progettare il tuo cluster di cache.

Note

Amazon ElastiCache funziona con entrambi i motori Redis e Memcached. Utilizza la guida relativa al motore che ti interessa. Se non sei certo di quale motore utilizzare, consulta [Confronto tra cache progettate autonomamente Memcached e Redis](#) in questa guida.

Cache serverless

ElastiCache for Memcached offre il caching senza server, che semplifica l'aggiunta e il funzionamento di una cache basata su Memcached per l'applicazione. ElastiCache per Memcached Serverless consente di creare una cache ad alta disponibilità in meno di un minuto ed elimina la necessità di effettuare il provisioning di istanze o configurare nodi o cluster. Gli sviluppatori possono creare una cache serverless specificando il nome della cache utilizzando la ElastiCache console, l'SDK o la CLI.

ElastiCache Serverless elimina inoltre la necessità di pianificare e gestire la capacità di memorizzazione nella cache. ElastiCache monitora costantemente la memoria e il calcolo della cache utilizzati dall'applicazione e ridimensiona automaticamente la capacità per soddisfare le esigenze dell'applicazione. ElastiCache offre un'esperienza endpoint semplice per gli sviluppatori, astruendo l'infrastruttura di cache e il software sottostanti. ElastiCache gestisce il provisioning dell'hardware, il monitoraggio, la sostituzione dei nodi e l'applicazione di patch software in modo automatico e trasparente, in modo da potersi concentrare sullo sviluppo delle applicazioni anziché utilizzare la cache.

ElastiCache per Memcached Serverless è compatibile con Memcached 1.6 e versioni successive.

Progettarne uno personalizzato per il cluster Memcached ElastiCache

Se hai bisogno di un controllo preciso sul tuo cluster ElastiCache for Memcached, puoi scegliere di progettare il tuo cluster Memcached con. ElastiCache consente di gestire un cluster basato su nodi, scegliendo il tipo di nodo, il numero di nodi e il posizionamento dei nodi nelle zone di disponibilità del cluster. AWS Trattandosi di un servizio completamente gestito, ElastiCache gestisce automaticamente il provisioning dell'hardware, il monitoraggio, la sostituzione dei nodi e l'applicazione di patch software per il cluster.

La progettazione personalizzata ElastiCache per il cluster Memcached offre maggiore flessibilità e controllo sui cluster. Ad esempio, puoi scegliere di gestire un cluster con disponibilità per singola zona di disponibilità o per più zone di disponibilità a seconda delle tue esigenze. Quando si progettano i propri cluster, è responsabilità dell'utente scegliere correttamente il tipo e il numero di nodi per garantire che la cache disponga della capacità sufficiente a soddisfare le esigenze dell'applicazione. È inoltre possibile scegliere quando applicare le nuove patch del software al cluster Memcached.

Quando si progetta un proprio programma ElastiCache per Memcached, è possibile scegliere di eseguire Memcached 1.4 e versioni successive.

Come funziona

Qui puoi trovare una panoramica dei componenti principali di una distribuzione ElastiCache per Memcached.

Motori di cache e caching

Una cache è un archivio dati in memoria che è possibile utilizzare per archiviare i dati memorizzati nella cache. In genere, l'applicazione memorizza nella cache i dati a cui si accede di frequente per ottimizzare i tempi di risposta. ElastiCache for Memcached offre due opzioni di implementazione: cluster serverless e cluster progettati autonomamente. Per informazioni, consultare [Scelta tra le opzioni di implementazione](#).

Note

Amazon ElastiCache funziona con entrambi i motori Redis e Memcached. Utilizza la guida relativa al motore che ti interessa. Se non sei certo di quale motore utilizzare, consulta [Confronto tra cache progettate autonomamente Memcached e Redis](#) in questa guida.

Argomenti

- [Come funziona Memcached ElastiCache](#)
- [Prezzi delle dimensioni](#)

Come funziona Memcached ElastiCache

ElastiCache per Memcached Serverless

ElastiCache per Memcached Serverless consente di creare una cache senza preoccuparsi della pianificazione della capacità, della gestione dell'hardware o della progettazione del cluster. È sufficiente fornire il nome della cache per ricevere un singolo endpoint che puoi configurare nel client Memcached per iniziare ad accedere alla cache.

Note

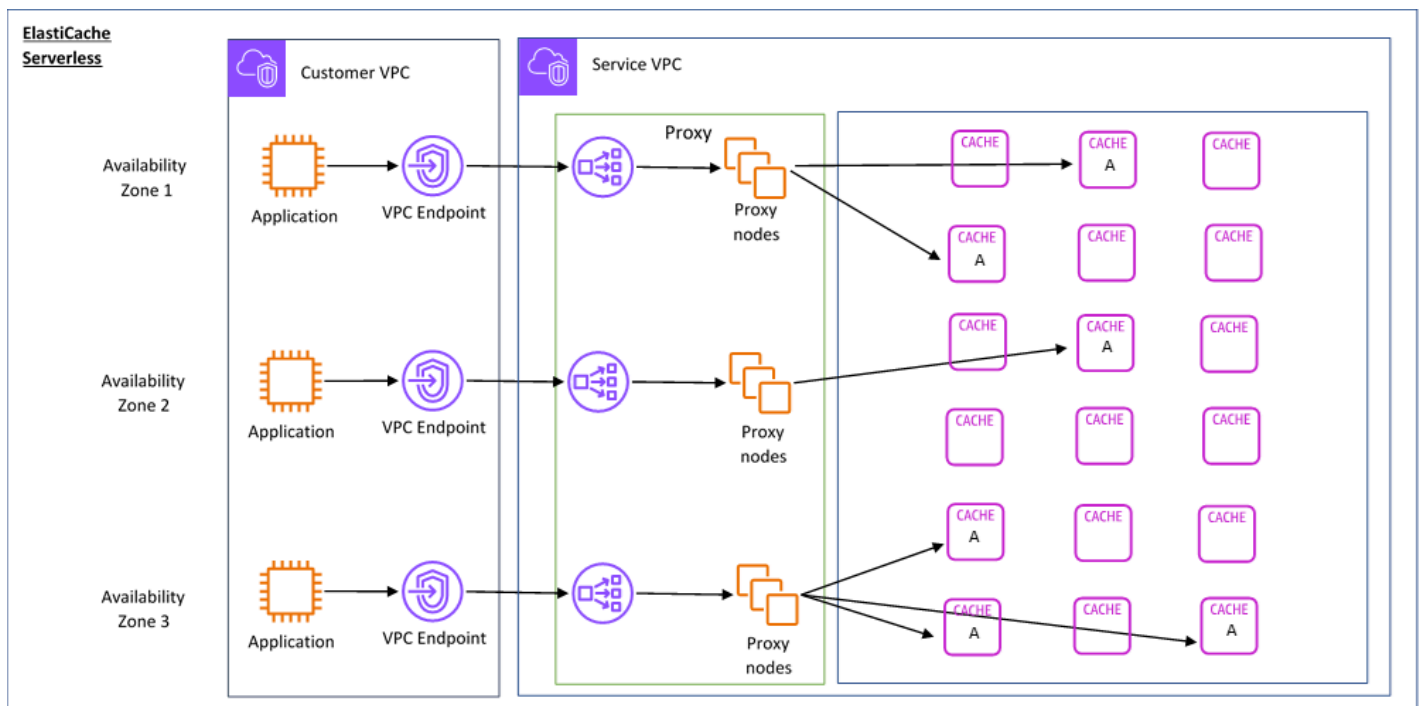
ElastiCache per Memcached Serverless è compatibile solo con i client Memcached che supportano TLS.

Vantaggi principali

- Nessuna pianificazione della capacità: ElastiCache Serverless elimina la necessità di pianificare la capacità. ElastiCache Serverless monitora continuamente la memoria, l'elaborazione e l'utilizzo della larghezza di banda di rete della cache e scala sia verticalmente che orizzontalmente. Consente di aumentare le dimensioni di un nodo di cache, avviando parallelamente un'operazione di scalabilità orizzontale per garantire che la cache possa dimensionarsi per soddisfare i requisiti dell'applicazione in ogni momento.
- Ray-per-use: Con ElastiCache Serverless, paghi per i dati archiviati e di calcolo utilizzati dal carico di lavoro sulla cache. Per informazioni, consulta [Prezzi delle dimensioni](#).

- **Alta disponibilità:** ElastiCache Serverless replica automaticamente i dati su più zone di disponibilità (AZ) per un'elevata disponibilità. Monitora automaticamente i nodi di cache sottostanti e li sostituisce in caso di errore. Offre uno SLA con disponibilità del 99,99% per ogni cache.
- **Aggiornamenti automatici del software:** ElastiCache Serverless aggiorna automaticamente la cache all'ultima versione del software secondario e all'ultima versione del software con patch senza alcun impatto sulla disponibilità dell'applicazione. Quando sarà disponibile una nuova versione principale di Memcached, ElastiCache ti invierà una notifica.
- **Sicurezza:** serverless esegue sempre la crittografia dei dati in transito e a riposo. È possibile utilizzare una chiave gestita dal servizio o la chiave gestita dal cliente per crittografare i dati a riposo.

Il diagramma seguente illustra come funziona Serverless. ElastiCache



Quando crei una nuova cache serverless, ElastiCache crea un endpoint Virtual Private Cloud (VPC) nelle sottoreti di tua scelta nel tuo VPC. L'applicazione può connettersi alla cache tramite questi endpoint VPC.

Con ElastiCache Serverless ricevi un singolo endpoint DNS a cui si connette l'applicazione. Quando richiedi una nuova connessione all'endpoint, ElastiCache Serverless gestisce tutte le connessioni alla cache tramite un livello proxy. Il livello del proxy aiuta a ridurre la complessità della configurazione del client perché il client non ha bisogno di individuare nuovamente la topologia del cluster in

caso di modifiche al cluster sottostante. Il livello del proxy è un set di nodi proxy che gestiscono le connessioni utilizzando un Network Load Balancer. Quando l'applicazione crea una nuova connessione alla cache, la richiesta viene inviata a un nodo proxy dal Network Load Balancer. Quando l'applicazione esegue i comandi della cache, il nodo proxy connesso all'applicazione elabora le richieste su un nodo di cache. Il livello del proxy estrae la topologia e i nodi del cluster di cache dal client. Ciò consente di ElastiCache bilanciare in modo intelligente il carico, scalare e aggiungere nuovi nodi di cache, sostituire i nodi di cache in caso di guasto e aggiornare il software sui nodi di cache, il tutto senza influire sulla disponibilità sull'applicazione o dover reimpostare le connessioni.

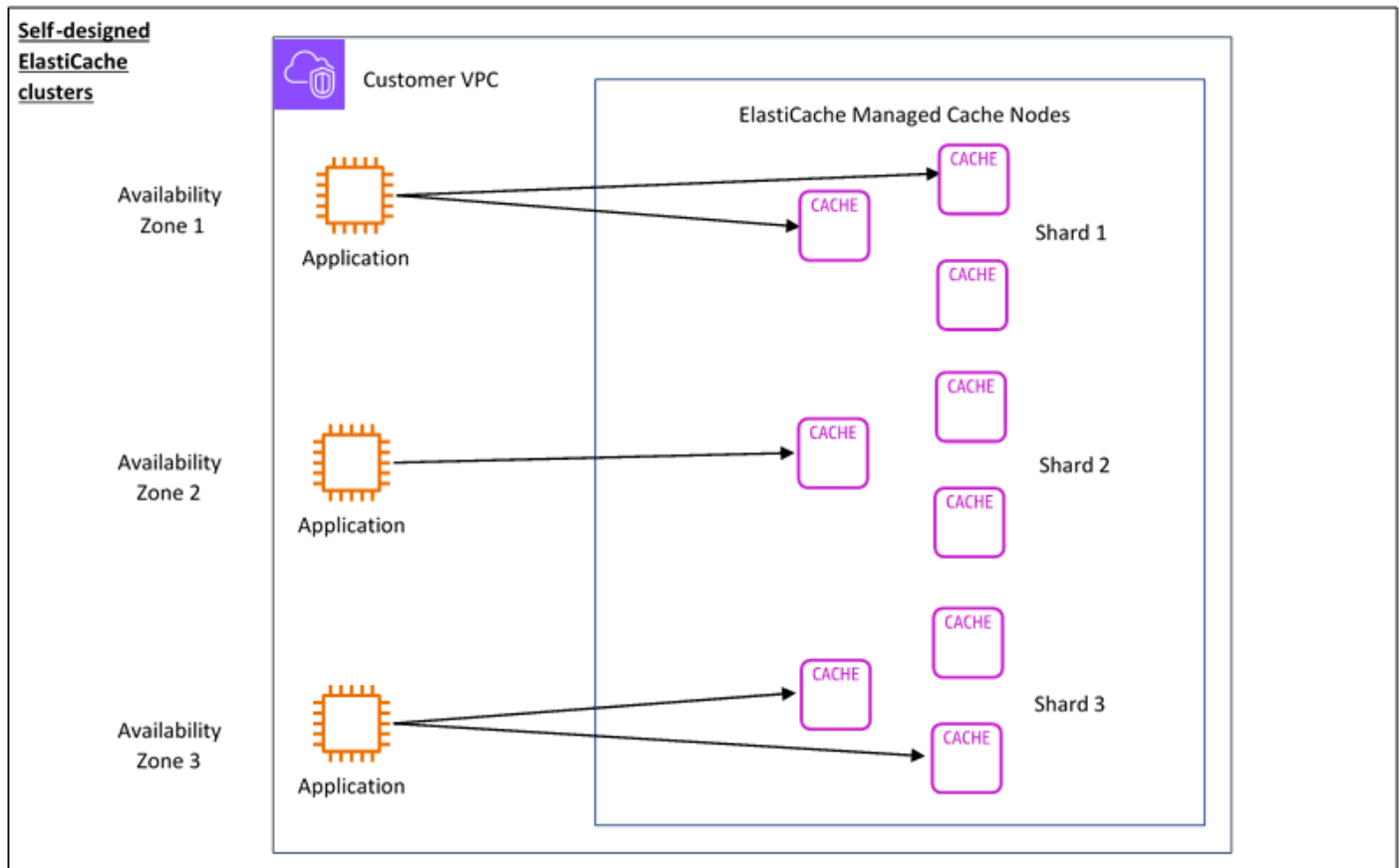
Cluster progettati autonomamente ElastiCache

Puoi scegliere di progettare i tuoi ElastiCache cluster scegliendo una famiglia di nodi di cache, le dimensioni e il numero di nodi per il cluster. La progettazione del proprio cluster offre un controllo più granulare sulla configurazione e sul dimensionamento del cluster.

Vantaggi principali

- **Progetta il tuo cluster:** con ElastiCache, puoi progettare il tuo cluster e scegliere dove posizionare i nodi di cache. Ad esempio, se un'applicazione compromette l'alta disponibilità con una bassa latenza, puoi scegliere di implementare i nodi di cache in un'unica zona di disponibilità. In alternativa, puoi progettare il tuo cluster con nodi su più zone di disponibilità per ottenere l'alta disponibilità.
- **Controllo granulare:** quando progetti il tuo cluster, hai un maggiore controllo sull'ottimizzazione delle impostazioni della cache. Ad esempio, puoi utilizzare [Configurazione dei parametri di motore con i gruppi di parametri](#) per configurare il motore di cache.
- **Dimensionamento verticale e orizzontale:** puoi scegliere di dimensionare manualmente il cluster aumentando o diminuendo la dimensione del nodo di cache quando necessario. Puoi anche dimensionare orizzontalmente aggiungendo i nodi.

Il diagramma seguente illustra come funzionano i cluster ElastiCache progettati autonomamente.



Prezzi delle dimensioni

È possibile eseguire la distribuzione in due opzioni di implementazione. ElastiCache Quando si implementa ElastiCache Serverless, si paga in base all'utilizzo dei dati archiviati in GB all'ora e per l'elaborazione in unità di elaborazione (ECPU). ElastiCache Quando si sceglie di progettare autonomamente i cluster Memcached, si ElastiCache paga in base all'ora di utilizzo del nodo di cache. Per i dettagli dei prezzi, consulta [questa pagina](#).

Archiviazione di dati

Paghi per i dati archiviati in ElastiCache Serverless e fatturati in gigabyte/ora (GB-h). ElastiCache Serverless monitora continuamente i dati archiviati nella cache, campionandoli più volte al minuto, e calcola una media oraria per determinare l'utilizzo dell'archiviazione dei dati della cache in GB-h. Ogni cache ElastiCache Serverless viene misurata per un minimo di 1 GB di dati archiviati.

ElastiCache Unità di elaborazione (ECPU)

Paghi per le richieste eseguite dalla tua applicazione su ElastiCache Serverless in ElastiCache Processing Units (ECPU), un'unità che include sia il tempo di vCPU che i dati trasferiti.

- Le operazioni di lettura e scrittura semplici richiedono 1 ECPU per ogni kilobyte (KB) di dati trasferiti. Ad esempio, un comando GET che trasferisce fino a 1 KB di dati consuma 1 ECPU. Una richiesta SET che trasferisce 3,2 KB di dati consuma 3,2 ECPU.
- I comandi che operano su più elementi consumano proporzionalmente più ECPU. Ad esempio, se l'applicazione esegue un comando multiget su 3 elementi, consuma 3 ECPU.
- I comandi che operano su più elementi e trasferiscono più dati consumano le ECPU in base alla dimensione più alta tra le due. Ad esempio, se l'applicazione utilizza il comando GET, recupera 3 elementi e trasferisce 3,2 KB di dati, consuma 3,2 ECPU. In alternativa, se trasferisce solo 2 KB di dati, consuma 3 ECPU.

ElastiCache Serverless emette una nuova metrica chiamata `ElastiCacheProcessingUnits` che ti aiuta a comprendere le ECPU utilizzate dal tuo carico di lavoro.

Ore per nodo

Puoi scegliere di progettare il tuo cluster di cache scegliendo la famiglia di nodi EC2, la dimensione, il numero di nodi e il posizionamento nelle zone di disponibilità. Quando si progetta autonomamente il cluster, si paga per ora di utilizzo per ogni nodo di cache.

Casi d'uso comuni relativi a ElastiCache e in che modo ElastiCache può essere d'aiuto

Che si tratti di fornire le ultime notizie del giorno o un catalogo di prodotti o che si tratti di vendere biglietti per un evento, la velocità è alla base di ogni operazione. Il successo del tuo sito web e del tuo business è influenzato significativamente dalla velocità con cui si distribuiscono i contenuti.

Nell'articolo "[For Impatient Web Users, an Eye Blink Is Just Too Long to Wait \(Per gli utenti del web impazienti, attendere un battito di ciglia è troppo\)](#)", il New York Times ha stimato che gli utenti possono registrare una differenza di 250 millisecondi (1/4 di secondo) tra i siti concorrenti. Gli utenti tendono a scartare il sito più lento a favore di quello più veloce. Test effettuati da Amazon, secondo quanto descritto in [How Webpage Load Time Is Related to Visitor Loss \(Come il tempo di caricamento di una pagina Web è collegato alla perdita di visitatori\)](#), hanno rivelato che ogni 100 ms in più (1/10 di secondo) impiegati nel caricamento della pagina, le vendite subiscono una flessione dell'1 per cento.

Se qualcuno desidera dei dati, è possibile fornirli in modo molto più veloce se sono memorizzati nella cache. Che si tratti di una pagina web o di un report alla base di decisioni aziendali. La tua azienda può permettersi di non memorizzare in cache le pagine web in modo da distribuirle con la minore latenza possibile?

Intuitivamente, potrebbe essere piuttosto scontato pensare di memorizzare in cache gli elementi richiesti con maggiore frequenza. Invece, perché non memorizzare in cache gli elementi richiesti con minore frequenza? Anche le chiamate API remote o le query di database meglio ottimizzate sono notevolmente più lente rispetto a quando si richiama una chiave flat da una cache in memoria. I tempi notevolmente più lenti spingono i clienti ad andare altrove.

I seguenti esempi mostrano alcuni dei modi in cui l'utilizzo di ElastiCache può migliorare le prestazioni generali dell'applicazione.

Datastore in memoria

Lo scopo principale di uno store chiave-valore in memoria è fornire accesso ultrarapido (latenza in millisecondi) e a costo zero alle copie di dati. La maggior parte dei datastore hanno aree di dati a cui si accede di frequente ma che vengono aggiornate raramente. Inoltre, l'interrogazione di un database è sempre più lenta e più costosa dell'individuazione di una chiave in una cache della coppia chiave-valore. Alcune query di database sono particolarmente costose da eseguire. Ad esempio, le query che implicano unioni di più tabelle o le query con calcoli complessi. Memorizzando nella cache

tali risultati delle query, si paga il prezzo della query una sola volta. Quindi è possibile richiamare velocemente i dati più volte senza dover rieseguire la query.

Cosa devo memorizzare nella cache?

Quando si decide quali dati memorizzare nella cache, tenere presenti i seguenti fattori:

Velocità e spese – È sempre più lento e costoso ottenere dati da un database che da una cache. Alcune query del database sono per natura più lente e più costose di altre. Ad esempio, le query che eseguono unioni su più tabelle sono molto più lente e più costose delle semplici query a tabella unica. Se l'acquisizione di dati interessanti richiede una query più lenta e più costosa, è indicata per il caching. Se l'acquisizione di dati richiede una query semplice e relativamente rapida, può essere idonea per il caching ma è necessario valutare altri fattori.

Dati e modello di accesso— Determinare cosa memorizzare nella cache comporta anche la comprensione dei dati stessi e dei relativi modelli di accesso. Ad esempio, non ha senso memorizzare nella cache i dati sottoposti a continue modifiche o a cui si accede raramente. Affinché la memorizzazione nella cache fornisca vantaggi reali, i dati devono essere relativamente statici e l'accesso a essi deve essere frequente. Un esempio è un profilo personale su un sito di social media. D'altra parte, non è necessario memorizzare nella cache se questa operazione non fornisce vantaggi economici e non migliora la velocità. Ad esempio, non ha senso memorizzare in cache le pagine web che restituiscono risultati di ricerca, perché le query e i risultati sono di solito univoci.

Obsolescenza – Per definizione, i dati memorizzati nella cache sono dati obsoleti. Anche se in determinate circostanze non sono obsoleti, dovrebbero sempre essere considerati e trattati come obsoleti. Per dire se i dati siano candidati al caching, devi stabilire la tolleranza dell'applicazione per i dati obsoleti.

L'applicazione potrebbe essere in grado di tollerare i dati obsoleti in un dato contesto, ma non in un altro. Supponi, ad esempio, che il sito fornisca un prezzo delle azioni quotato in borsa. I clienti potrebbero accettare una certa obsolescenza con una dichiarazione di non responsabilità secondo la quale i prezzi potrebbero essere n minuti in ritardo. Tuttavia, quando si offre il prezzo per lo stesso stock a un broker che effettua una vendita o un acquisto, sono necessari i dati in tempo reale.

Prendi in considerazione di memorizzare nella cache i dati in presenza di una delle seguenti condizioni:

- L'acquisizione dei dati è lenta o costosa se confrontata con il recupero dalla cache.
- Gli utenti accedono spesso ai dati.

- I dati rimangono relativamente gli stessi, o se cambiano rapidamente l'obsolescenza non è un grosso problema.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- [Strategie di caching](#) nella Guida per l'utente di ElastiCache for Memcached

Testimonianze dei clienti su ElastiCache

Per scoprire come aziende del calibro di Airbnb, PBS, Esri e molte altre utilizzano Amazon ElastiCache per far crescere il loro business con un'esperienza cliente potenziata, consulta [Modi di utilizzo di ElastiCache da parte dei clienti](#).

Puoi anche guardare i [video tutorial](#) per altri casi d'uso dei clienti ElastiCache.

Risorse di Memcached ElastiCache

Ti consigliamo di iniziare a leggere le seguenti sezioni e di farvi riferimento ogni volta che ti servono:

- Caratteristiche salienti del servizio e prezzi - Nella [pagina di dettaglio del prodotto](#) trovi la panoramica prodotto, i punti salienti del servizio e i prezzi di ElastiCache.
- ElastiCache Video— Il [Video relativi a ElastiCache](#) contiene video che introducono ad Amazon ElastiCache for Memcached, coprono casi d'uso comuni e dimostrano come utilizzare ElastiCache for MemCached per ridurre la latenza e migliorare la velocità effettiva delle applicazioni.
- Nozioni di base: la sezione [Nozioni di base Amazon ElastiCache for Memcached](#) include un esame che illustra il processo di creazione di un cluster di cache, l'autorizzazione dell'accesso al cluster di cache, la connessione a un nodo di cache e l'eliminazione del cluster di cache.
- Prestazioni su scala— Le [Prestazioni su scala con il white paper di Amazon ElastiCache](#) affrontano le strategie di memorizzazione nella cache che consentono all'applicazione di funzionare bene su larga scala.

Se vuoi utilizzare l'AWS Command Line Interface (AWS CLI), puoi utilizzare questi documenti per iniziare:

- [Documentazione di AWS Command Line Interface](#)

In questa sezione vengono fornite informazioni sul download della AWS CLI, come fare in modo che la AWS CLI lavori sul tuo sistema e fornendo le tue credenziali AWS.

- [Documentazione relativa alla AWS CLI per ElastiCache](#)

Questo documento distinto riguarda tutti i comandi della AWS CLI per ElastiCache, inclusi la sintassi e gli esempi.

Puoi scrivere programmi applicativi per utilizzare l'API di ElastiCache con una serie di linguaggi di programmazione di uso comune. Ecco alcune risorse:

- [Strumenti per Amazon Web Services](#)

Amazon Web Services offre numerosi Software Development Kit (SDK) che supportano ElastiCache for Memcached. Puoi scrivere codici per ElastiCache utilizzando Java, .NET, PHP, Ruby e altri linguaggi. Questi SDK possono semplificare enormemente lo sviluppo delle applicazioni formattando le richieste a ElastiCache, analizzando le risposte e offrendo logica di ripetizione e gestione degli errori.

- [Utilizzo dell'API ElastiCache](#)

Se non intendi utilizzare i SDK di AWS, puoi interagire direttamente con ElastiCache utilizzando l'API di query. In questa sezione puoi trovare suggerimenti per risolvere i problemi e informazioni su come creare e autenticare richieste e gestire risposte.

- [Informazioni sull'API di Amazon ElastiCache](#)

Questo documento distinto riguarda tutte le operazioni dell'API di ElastiCache, inclusi la sintassi e gli esempi.

Strumenti per la gestione dell'implementazione

Una volta concesso all'istanza Amazon EC2 l'accesso al cluster ElastiCache, hai quattro opzioni per gestire il cluster : la AWS Management Console, la AWS CLI di ElastiCache, il AWS SDK per ElastiCache e l'API ElastiCache.

Utilizzo di AWS Management Console

La AWS Management Console è il modo più semplice per gestire Amazon ElastiCache for Memcached. La console ti permette di creare cluster di cache, aggiungere e rimuovere nodi di cache ed eseguire altre attività amministrative senza dover scrivere codici. La console offre anche grafici sulle prestazioni dei nodi di cache da CloudWatch, che mostrano l'attività del motore della cache, l'utilizzo della memoria e della CPU, nonché altri parametri. Per ulteriori informazioni, consulta gli argomenti specifici nella Guida per l'utente.

Utilizzo di AWS CLI

È possibile utilizzare anche la (AWS Command Line Interface) AWS CLI per ElastiCache. L'AWS CLI semplifica l'esecuzione di operazioni successive, ad esempio l'avvio e l'arresto del cluster di cache. Puoi anche invocare la AWS CLI per i comandi di ElastiCache da un linguaggio di scripting di tua scelta, permettendoti di automatizzare le attività ripetute. Per ulteriori informazioni sulla AWS CLI, consulta la Guida per l'utente la [AWS CLI Documentazione di riferimento sui comandi della](#) .

Utilizzo dell'SDK AWS

Se vuoi accedere a ElastiCache da un'applicazione, puoi utilizzare uno dei Software Development Kit (SDK) AWS. I SDK eseguono il wrapping delle chiamate API di ElastiCache e isolano la tua applicazione dai dettagli di basso livello dell'API di ElastiCache. Basta fornire le credenziali e le librerie del kit SDK eseguono automaticamente l'autenticazione e la firma delle richieste. Per ulteriori informazioni sull'utilizzo di altri SDK AWS, consulta [Strumenti per Amazon Web Services](#).

Utilizzo dell'API ElastiCache

Puoi anche scrivere il codice dell'applicazione direttamente sull'API del servizio Web ElastiCache. Quando utilizzi l'API, devi scrivere il codice necessario per costruire e autenticare le richieste HTTP, analizzare i risultati provenienti da ElastiCache e gestire gli eventuali errori. Per ulteriori informazioni sull'API, consulta [Utilizzo dell'API ElastiCache](#).

Consulta anche

Per informazioni più dettagliate sulla gestione del Amazon ElastiCache per la distribuzione di Memcached, consulta i seguenti argomenti:

- [Lavorare con ElastiCache](#)
- [Riservatezza del traffico Internet](#)
- [Registrazione di log e monitoraggio in Amazon ElastiCache](#)

Scelta tra le opzioni di implementazione

Amazon ElastiCache offre due opzioni di implementazione:

- Cache serverless
- Progetta il tuo cluster

Cache serverless

Amazon ElastiCache serverless semplifica la creazione di cache e si dimensiona istantaneamente per supportare le applicazioni più esigenti dei clienti. Con ElastiCache serverless, puoi creare una cache altamente disponibile e dimensionabile in meno di un minuto, eliminando la necessità di allocare, pianificare e gestire la capacità del cluster di cache. ElastiCache serverless archivia automaticamente i dati in modo ridondante su tre zone di disponibilità e fornisce un [Accordo sul livello di servizio \(SLA\)](#) con disponibilità del 99,99%. ElastiCache fornisce la replica automatica dei dati tra le zone di disponibilità eliminando la necessità di gestire manualmente le repliche e il software personalizzato per mantenerle sincronizzate.

Cluster progettati autonomamente

Se hai bisogno di un controllo granulare sul cluster ElastiCache per Memcached, puoi scegliere di progettare il tuo cluster Memcached con ElastiCache. ElastiCache consente di usare un cluster basato su nodi, scegliendo il tipo di nodo, il numero di nodi e il posizionamento dei nodi nelle zone di disponibilità AWS del cluster. Poiché ElastiCache è un servizio completamente gestito, automaticamente fornisce il provisioning dell'hardware, il monitoraggio, la sostituzione dei nodi e l'applicazione delle patch al software del cluster.

Scelta tra le opzioni di implementazione

Scegli la cache serverless se:

- crei una nuova cache per carichi di lavoro nuovi o sconosciuti;
- il traffico delle applicazioni non è prevedibile;
- desideri iniziare a usare una cache nel modo più semplice.

Scegli di progettare il tuo cluster ElastiCache se:

- stai già utilizzando ElastiCache serverless e desideri un controllo più preciso sul tipo di nodo che esegue Memcached, sul numero di nodi e sul posizionamento dei nodi;
- non ti aspetti che il traffico delle applicazioni oscilli molto oppure puoi prevedere con precisione il traffico massimo e minimo delle applicazioni;
- puoi prevedere i requisiti di capacità per controllare i costi.

Argomenti correlati:

- [Progettazione e gestione del proprio cluster ElastiCache per l'implementazione Memcached](#)

Confronto tra cache progettate autonomamente Memcached e Redis

Amazon ElastiCache supporta i motori cache Memcached e Redis. Ogni motore offre alcuni vantaggi. Utilizza le informazioni in questo argomento per individuare il motore e la versione che meglio si adattano alle tue esigenze.

Important

Dopo aver creato un cluster di cache o un gruppo di replica, puoi aggiornare a una versione del motore più recente, ma non puoi eseguire il downgrade a versioni motore precedenti. Se vuoi utilizzare una versione del motore precedente, elimina il cluster di cache o il gruppo di replica esistente e crealo di nuovo con la versione del motore precedente.

A prima vista, i motori sembrano simili. Ognuno di essi è uno store chiave-valore in memoria. Tuttavia, vi sono alcune differenze importanti.

Scegliere Memcached per le seguenti esigenze:

- Hai bisogno del modello in assoluto più semplice.
- Hai necessità di eseguire grandi nodi con più core o thread.
- Hai la possibilità di dimensionare, aggiungere e rimuovere i nodi a seconda che la domanda del tuo sistema sia crescente o decrescente.
- Hai necessità di memorizzare gli oggetti nella cache.

Scegliere Redis con una versione di ElastiCache se si applicano le seguenti condizioni:

- ElastiCache per Redis versione 7.0 (avanzata)

Si desidera utilizzare [Funzioni Redis](#), [Pub/Sub partizionato](#) o [Miglioramenti ACL Redis](#). Per ulteriori informazioni, consultare [Redis Versione 7.0 \(avanzata\)](#).

- ElastiCache for Redis versione 6.2 (avanzata)

Vuoi la possibilità di eseguire il livello dei dati tra memoria e SSD utilizzando il tipo di nodo r6gd. Per ulteriori informazioni, consulta la sezione [Tiering di dati](#).

- ElastiCache for Redis versione 6.0 (avanzata)

Vuoi autenticare gli utenti con il controllo accessi basato sui ruoli.

Per ulteriori informazioni, consulta la sezione [Redis Versione 6.0 \(avanzata\)](#).

- ElastiCache for Redis versione 5.0.0 (potenziata)

Vuoi utilizzare i [flussi Redis](#), una struttura di registro dei dati che consente ai produttori di accodare nuove voci in tempo reale e che consente ai consumatori di utilizzare messaggi con blocchi o senza blocchi.

Per ulteriori informazioni, consulta la sezione [Redis versione 5.0.0 \(potenziata\)](#).

- ElastiCache for Redis versione 4.0.10 (potenziata)


Supporta la crittografia e l'aggiunta o la rimozione automatica degli shard dal cluster Redis (modalità cluster abilitata).

Per ulteriori informazioni, consulta la sezione [Redis versione 4.0.10 \(potenziata\)](#).

Le seguenti versioni sono obsolete, hanno raggiunto o stanno per giungere a fine vita.

- ElastiCache for Redis versione 3.2.10 (potenziata)

Supporta la capacità di aggiungere o rimuovere dinamicamente gli shard dal cluster Redis (modalità cluster abilitata).

 Important

Al momento ElastiCache for Redis 3.2.10 non supporta la crittografia.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- [Redis versione 3.2.10 \(potenziata\)](#)
 - Best practice per il resharding online di Redis, per ulteriori informazioni consultare:
 - [Best practice: resharding online](#)
 - [Resharding online e ribilanciamento partizioni per Redis \(modalità cluster abilitata\)](#)
 - Per ulteriori informazioni sul dimensionamento dei cluster Redis, consulta [Dimensionamento](#).
- ElastiCache for Redis versione 3.2.6 (potenziata)

Se sono necessarie la caratteristiche delle versioni Redis precedenti e le caratteristiche seguenti, scegliere ElastiCache for Redis 3.2.6:

- Crittografia dei dati in transito Per ulteriori informazioni consulta [Crittografia dei dati in transito di Amazon ElastiCache for Redis](#).
 - Crittografia dei dati inattivi Per ulteriori informazioni, consulta [Crittografia dei dati inattivi di Amazon ElastiCache per Redis](#).
- ElastiCache for Redis (modalità cluster abilitata) versione 3.2.4

Se sono necessarie la caratteristiche delle versioni Redis 2.8.x e le caratteristiche seguenti, scegliere Redis 3.2.4 (modalità cluster):

- Hai la necessità di eseguire la partizione dei dati tra due fino a 500 gruppi di nodi (solo in modalità cluster).
 - Hai la necessità di effettuare l'indicizzazione geospaziale (modalità cluster o non-cluster).
 - Non è necessario il support di più database.
- ElastiCache for Redis (modalità non cluster) 2.8.x e 3.2.4 (potenziata)

- Sono necessari tipi di dati complessi, ad esempio stringa, set, set ordinati, elenchi, hash e bitmap.
- È necessario ordinare o classificare i dataset in-memory.
- Il tuo store di chiavi deve essere permanente.
- È necessario replicare i dati dai primari a quelli relativi a una o più repliche di lettura per applicazioni di letture intensive.
- Se il primo nodo riporta un errore, è necessario un failover automatico.
- È necessario pubblicare e sottoscrivere le caratteristiche(pub/sub) per informare i clienti sugli eventi sul server.
- Sono necessarie caratteristiche di backup e ripristino.
- È necessario il support di più database.

Riepilogo del confronto di Memcached, Redis (modalità cluster disabilitata) e Redis (modalità cluster abilitata)

	Memcached	Redis (modalità cluster disabilitata)	Redis (modalità cluster abilitata)
Engine versions+	1.4.5 and later	4.0.10 and later	4.0.10 and later
Data types	Simple	2.8.x - Complex * Complex	3.2.x and later - Complex
Data partitioning	Yes	No	Yes
Cluster is modifiable	Yes	Yes	3.2.10 and later - Limited
Online resharding	No	No	3.2.10 and later
Encryption	in-transit 1.6.12 and later	4.0.10 and later	4.0.10 and later
Data tiering	No	6.2 and later	6.2 and later
Certificazioni di conformità			
Compliance Certification			
FedRAMP	Sì - 1.6.12 e versioni successive	versioni 4.0.10 e successive	versioni 4.0.10 e successive
HIPAA	Sì - 1.6.12 e versioni successive	versioni 4.0.10 e successive	versioni 4.0.10 e successive
PCI DSS	Sì	versioni 4.0.10 e successive	versioni 4.0.10 e successive
Multi-threaded	Yes	No	No
Node type upgrade	No	Yes	Yes
Engine upgrading	Yes	Yes	Yes

	Memcached	Redis (modalità cluster disabilitata)	Redis (modalità cluster abilitata)
High availability (replication)	No	Yes	Yes
Automatic failover	No	Optional	Required
Pub/Sub capabilities	No	Yes	Yes
Sorted sets	No	Yes	Yes
Backup and restore	No	Yes	Yes
Geospatial indexing	No	4.0.10 and later	Yes

Note:

string, objects (like databases)

* string, sets, sorted sets, lists, hashes, bitmaps, hyperloglog

string, sets, sorted sets, lists, hashes, bitmaps, hyperloglog, geospatial indexes

+ Excludes versions which are deprecated, have reached or soon to reach end of life.

Dopo aver scelto il motore per il cluster, ti consigliamo di utilizzare la versione più recente per quel motore. Per ulteriori informazioni, consulta [Versioni supportate di ElastiCache for Memcached](#) o [Versioni supportate di ElastiCache for Redis](#).

Nozioni di base Amazon ElastiCache for Memcached

Gli argomenti in questa sezione illustrano il processo di creazione, autorizzazione dell'accesso, connessione e infine eliminazione di una cache serverless Memcached utilizzando la console ElastiCache.

Argomenti

- [Configurazione](#)
- [Fase 1. Creazione di una cache](#)
- [Fase 2: lettura e scrittura di dati nella cache](#)
- [Fase 3: \(facoltativo\) pulizia](#)
- [Fasi successive](#)
- [Tutorial: Configurazione di una funzione Lambda per l'accesso ad Amazon ElastiCache in un Amazon VPC](#)
- [Esercitazioni e video relativi a ElastiCache](#)

Configurazione

Per configurare ElastiCache:

Argomenti

- [Registrati per un Account AWS](#)
- [Creazione di un utente amministratore](#)
- [Concessione dell'accesso programmatico](#)
- [Configura le tue autorizzazioni \(solo nuovi ElastiCache utenti\)](#)
- [Configurazione di EC2](#)
- [Assegnazione dell'accesso di rete di un gruppo di sicurezza Amazon VPC alla cache](#)

Registrati per un Account AWS

Se non ne hai uno Account AWS, completa i seguenti passaggi per crearne uno.

Per iscriverti a un Account AWS

1. Apri la pagina all'indirizzo <https://portal.aws.amazon.com/billing/signup>.
2. Segui le istruzioni online.

Nel corso della procedura di registrazione riceverai una telefonata, durante la quale sarà necessario inserire un codice di verifica attraverso la tastiera del telefono.

Quando ti iscrivi a un Account AWS, Utente root dell'account AWS viene creato un. L'utente root dispone dell'accesso a tutte le risorse e tutti i Servizi AWS nell'account. Come best practice di sicurezza, [assegna l'accesso amministrativo a un utente amministrativo](#) e utilizza solo l'utente root per eseguire [attività che richiedono l'accesso di un utente root](#).

AWS ti invia un'email di conferma dopo il completamento della procedura di registrazione. È possibile visualizzare l'attività corrente dell'account e gestire l'account in qualsiasi momento accedendo all'indirizzo <https://aws.amazon.com/> e selezionando Il mio account.

Creazione di un utente amministratore

Dopo la registrazione Account AWS, proteggi Utente root dell'account AWS AWS IAM Identity Center, abilita e crea un utente amministrativo in modo da non utilizzare l'utente root per le attività quotidiane.

Proteggi i tuoi Utente root dell'account AWS

1. Accedi [AWS Management Console](#) come proprietario dell'account scegliendo Utente root e inserendo il tuo indirizzo Account AWS email. Nella pagina successiva, inserisci la password.

Per informazioni sull'accesso utilizzando un utente root, consulta la pagina [Accesso come utente root](#) della Guida per l'utente di Accedi ad AWS .

2. Abilita l'autenticazione a più fattori (MFA) per l'utente root.

Per istruzioni, consulta [Abilitare un dispositivo MFA virtuale per l'utente Account AWS root \(console\)](#) nella Guida per l'utente IAM.

Creazione di un utente amministratore

1. Abilita Centro identità IAM.

Per istruzioni, consulta [Abilitazione di AWS IAM Identity Center](#) nella Guida per l'utente di AWS IAM Identity Center .

- In Centro identità AWS IAM, assegna l'accesso amministrativo a un utente amministrativo.

Per un tutorial sull'utilizzo di IAM Identity Center directory come fonte di identità, consulta [Configurare l'accesso utente con l'impostazione predefinita IAM Identity Center directory](#) nella Guida per l'AWS IAM Identity Center utente.

Accesso come utente amministratore

- Per accedere con l'utente IAM Identity Center, utilizza l'URL di accesso che è stato inviato al tuo indirizzo e-mail quando hai creato l'utente IAM Identity Center.

Per informazioni sull'accesso utilizzando un utente IAM Identity Center, consulta [AWS Accedere al portale di accesso](#) nella Guida per l'Accedi ad AWS utente.

Concessione dell'accesso programmatico

Gli utenti hanno bisogno di un accesso programmatico se vogliono interagire con l' AWS AWS Management Console esterno di. Il modo per concedere l'accesso programmatico dipende dal tipo di utente che accede. AWS

Per fornire agli utenti l'accesso programmatico, scegli una delle seguenti opzioni.

Quale utente necessita dell'accesso programmatico?	Per	Come
Identità della forza lavoro (Utenti gestiti nel centro identità IAM)	Utilizza credenziali temporanee e per firmare le richieste programmatiche agli AWS CLI AWS SDK o alle API. AWS	<p>Segui le istruzioni per l'interfaccia che desideri utilizzare.</p> <ul style="list-style-type: none"> Per la AWS CLI, consulta Configurazione dell'uso AWS IAM Identity Center nella Guida AWS CLI per l'utente. AWS Command Line Interface

Quale utente necessita dell'accesso programmatico?	Per	Come
		<ul style="list-style-type: none"> Per AWS SDK, strumenti e AWS API, consulta l'autenticazione IAM Identity Center nella Guida di riferimento agli AWS SDK e agli strumenti.
IAM	Utilizza credenziali temporanee e per firmare le richieste programmatiche agli SDK o alle API AWS CLI. AWS AWS	Segui le istruzioni in Uso delle credenziali temporanee con AWS risorse nella Guida per l'utente IAM.
IAM	(Non consigliato) Utilizza credenziali a lungo termine per firmare le richieste programmatiche agli AWS CLI AWS SDK o alle API. AWS	<p>Segui le istruzioni per l'interfaccia che desideri utilizzare.</p> <ul style="list-style-type: none"> Per la AWS CLI, consulta Autenticazione tramite credenziali utente IAM nella Guida per l'utente. AWS Command Line Interface Per gli AWS SDK e gli strumenti, consulta Autenticazione tramite credenziali a lungo termine nella Guida di riferimento agli SDK e agli AWS strumenti. Per le AWS API, consulta Gestione delle chiavi di accesso per gli utenti IAM nella Guida per l'utente IAM.

Argomenti correlati:

- [Cos'è IAM?](#) nella Guida per l'utente di IAM.
- [AWS Informazioni generali sulle credenziali di sicurezza](#). AWS

Configura le tue autorizzazioni (solo nuovi ElastiCache utenti)

Per fornire l'accesso, aggiungi autorizzazioni ai tuoi utenti, gruppi o ruoli:

- Utenti e gruppi in AWS IAM Identity Center:

Crea un set di autorizzazioni. Segui le istruzioni riportate nella pagina [Create a permission set](#) (Creazione di un set di autorizzazioni) nella Guida per l'utente di AWS IAM Identity Center .

- Utenti gestiti in IAM tramite un provider di identità:

Crea un ruolo per la federazione delle identità. Segui le istruzioni riportate nella pagina [Creating a role for a third-party identity provider \(federation\)](#) (Creazione di un ruolo per un provider di identità di terze parti [federazione]) nella Guida per l'utente di IAM.

- Utenti IAM:

- Crea un ruolo che l'utente possa assumere. Per istruzioni, consulta la pagina [Creating a role for an IAM user](#) (Creazione di un ruolo per un utente IAM) nella Guida per l'utente di IAM.
- (Non consigliato) Collega una policy direttamente a un utente o aggiungi un utente a un gruppo di utenti. Segui le istruzioni riportate nella pagina [Aggiunta di autorizzazioni a un utente \(console\)](#) nella Guida per l'utente di IAM.

Amazon ElastiCache crea e utilizza ruoli collegati ai servizi per fornire risorse e accedere ad altre AWS risorse e servizi per tuo conto. Per creare ElastiCache per te un ruolo collegato ai servizi, utilizza la policy -managed denominata AWS. AmazonElastiCacheFullAccess Per questo ruolo viene effettuato il provisioning preventivo con l'autorizzazione necessaria al servizio per creare un ruolo collegato ai servizi per tuo conto.

Potresti decidere di non utilizzare la policy predefinita e di utilizzare piuttosto una policy gestita in modo personalizzato. In questo caso, verificare di avere le autorizzazioni per chiamare `iam:createServiceLinkedRole` o per creare il ruolo collegato ai servizi ElastiCache.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- [Creare una nuova policy](#) (IAM)

- [Policy gestite da AWS per Amazon ElastiCache](#)
- [Utilizzo di ruoli collegati ai servizi per Amazon ElastiCache](#)

Configurazione di EC2

Dovrai configurare un'istanza EC2 da cui potrai connetterti alla cache.

- Se non disponi già di un'istanza EC2, le informazioni per configurare un'istanza EC2 sono disponibili in [Tutorial: Get-started-with-Amazon-EC2-Linux-instances](#).
- L'istanza EC2 deve trovarsi nello stesso VPC e avere le stesse impostazioni del gruppo di sicurezza della cache. Per impostazione predefinita, Amazon ElastiCache crea una cache nel tuo VPC predefinito e utilizza il gruppo di sicurezza predefinito. Per questo tutorial, verifica che la tua istanza EC2 si trovi nel VPC predefinito e disponga del gruppo di sicurezza predefinito.

Assegnazione dell'accesso di rete di un gruppo di sicurezza Amazon VPC alla cache

ElastiCache per Memcached utilizza le porte 11211 e 11212 per accettare i comandi Memcached. Per connettersi ed eseguire correttamente i comandi Memcached dall'istanza EC2, il gruppo di sicurezza deve consentire l'accesso a queste porte.

1. Accedi AWS Command Line Interface e apri la console [Amazon EC2](#).
2. Nel riquadro di navigazione, in Rete e sicurezza, scegli Gruppi di sicurezza.
3. Dall'elenco dei gruppi di sicurezza, scegli il gruppo di sicurezza per il VPC Amazon. A meno che tu non abbia creato un gruppo di sicurezza da ElastiCache utilizzare, questo gruppo di sicurezza verrà denominato predefinito.
4. Seleziona la scheda In entrata, quindi:
 - a. Scegli Modifica.
 - b. Scegli Aggiungi regola.
 - c. Nella colonna Tipo, scegli Regola TCP personalizzata.
 - d. Nella casella Intervallo porte digita 11211.
 - e. Nella casella Origine scegli Ovunque che ha l'intervallo di porte (0.0.0.0/0) in modo che qualunque istanza Amazon EC2 avviata in Amazon VPC possa connettersi alla cache.
 - f. Se utilizzi ElastiCache serverless, aggiungi un'altra regola scegliendo Aggiungi regola.

- g. Nella colonna Tipo, scegli Regola TCP personalizzata.
- h. Nella casella Intervallo porte digita 11212.
- i. Nella casella Origine scegli Ovunque che ha l'intervallo di porte (0.0.0.0/0) in modo che qualunque istanza Amazon EC2 avviata in Amazon VPC possa connettersi alla cache.
- j. Seleziona Salva

Fase 1. Creazione di una cache

La cache che stai per avviare verrà eseguita in un ambiente attivo e non in un ambiente di sperimentazione (sandbox). Vengono addebitati i costi standard di utilizzo di ElastiCache relativi alla cache finché non viene eliminata. L'addebito totale è minimo (in genere meno di un dollaro) se completi l'esercizio descritto qui in una sola seduta ed elimini la cache alla fine. Per ulteriori informazioni sulle tariffe di utilizzo di ElastiCache, consulta [Amazon ElastiCache](#).

Creazione di una cache serverless

AWS Management Console

Per creare una nuova cache tramite la console ElastiCache:

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nel riquadro di navigazione sul lato sinistro della console scegli Cache Memcached.
3. Sul lato destro della console scegli Crea cache Memcached.
4. In Impostazioni cache inserisci un nome. Facoltativamente, è possibile inserire una descrizione per la cache.
5. Lascia le impostazioni predefinite selezionate.
6. Fai clic su Crea per creare la cache.
7. Una volta che lo stato della cache è "ATTIVO", puoi iniziare a scrivere e leggere i dati nella cache.

Per creare una nuova cache mediante la AWS CLI

L'esempio della AWS CLI seguente crea una nuova cache utilizzando create-serverless-cache.

Linux

```
aws elasticache create-serverless-cache \  
  --serverless-cache-name CacheName \  
  --engine memcached
```

Windows

```
aws elasticache create-serverless-cache ^  
  --serverless-cache-name CacheName ^  
  --engine memcached
```

Il valore del campo Stato è impostato su CREATING.

Per verificare se ElastiCache ha terminato la creazione della cache, utilizza il comando `describe-serverless-caches`.

Linux

```
aws elasticache describe-serverless-caches --serverless-cache-name CacheName
```

Windows

```
aws elasticache describe-serverless-caches --serverless-cache-name CacheName
```

Dopo aver creato la nuova cache, procedi alla [Fase 2: lettura e scrittura di dati nella cache](#).

Fase 2: lettura e scrittura di dati nella cache

Questa sezione presuppone che sia stata creata un'istanza Amazon EC2 ed eseguita la connessione alla stessa. Per istruzioni su come procedere, consultata la [Guida alle operazioni di base di Amazon EC2](#).

Per impostazione predefinita, ElastiCache crea una cache nel VPC predefinito. Assicurati che anche l'istanza EC2 sia creata nel VPC predefinito, in modo da potersi connettere alla cache.

Ricerca dell'endpoint della cache

AWS Management Console

Per trovare l'endpoint della cache tramite la console ElastiCache:

1. Accedi alla AWS Management Console e apri la console di Amazon ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nel riquadro di navigazione sul lato sinistro della console scegli Cache Memcached.
3. Sul lato destro della console fai clic sul nome della cache che hai appena creato.
4. In Dettagli della cache individua e copia l'endpoint della cache.

AWS CLI

L'esempio della AWS CLI seguente mostra come trovare l'endpoint per la nuova cache utilizzando il comando `describe-serverless-caches`. Dopo aver eseguito il comando, cerca il campo "Endpoint".

Linux

```
aws elasticache describe-serverless-caches \  
  --serverless-cache-name CacheName
```

Windows

```
aws elasticache describe-serverless-caches ^  
  --serverless-cache-name CacheName
```

Connessione tramite OpenSSL

Per informazioni su come eseguire la connessione tramite OpenSSL, consulta [Crittografia in transito ElastiCache \(TLS\)](#).

Connessione tramite il client Memcached Java

Per informazioni su come eseguire la connessione tramite il client Memcached Java, consulta [Crittografia in transito ElastiCache \(TLS\)](#).

Connessione tramite il client Memcached PHP

```
<?php  
$cluster_endpoint = "mycluster.serverless.use1.cache.amazonaws.com";  
$server_port = 11211;  
  
/* Initialize a persistent Memcached client in TLS mode */  
$tls_client = new Memcached('persistent-id');
```

```
$tls_client->addServer($cluster_endpoint, $server_port);
if(!$tls_client->setOption(Memcached::OPT_USE_TLS, 1)) {
    echo $tls_client->getLastErrorMessage(), "\n";
    exit(1);
}
$tls_config = new MemcachedTLSContextConfig();
$tls_config->hostname = '*.serverless.us1.cache.amazonaws.com';
$tls_config->skip_cert_verify = false;
$tls_config->skip_hostname_verify = false;
$tls_client->createAndSetTLSContext((array)$tls_config);

/* store the data for 60 seconds in the cluster */
$tls_client->set('key', 'value', 60);
?>
```

Connessione tramite il client Memcached Python (Pymemcache)

Consulta https://pymemcache.readthedocs.io/en/latest/getting_started.html

```
import ssl
from pymemcache.client.base import Client

context = ssl.create_default_context()
cluster_endpoint = <To be taken from the AWS CLI / console>
target_port = 11211
memcached_client = Client("{cluster_endpoint}", target_port, tls_context=context)
memcached_client.set("key", "value", expire=500, noreply=False)
assert self.memcached_client.get("key").decode() == "value"
```

Connessione tramite il client Memcached NodeJS/TS (Electrode-IO memcache)

Consulta <https://github.com/electrode-io/memcache> e <https://www.npmjs.com/package/memcache-client>

Installa usando npm i memcache-client.

Nell'applicazione, crea un client TLS Memcached come segue:

```
var memcache = require("memcache-client");
const client = new memcache.MemcacheClient({server: "{cluster_endpoint}:11211", tls:
  {}});
client.set("key", "value");
```

Connessione tramite il client Memcached Rust (rust-memcache)

Consulta <https://crates.io/crates/memcache> e <https://github.com/aisk/rust-memcache>.

```
// create connection with to memcached server node:
let client = memcache::connect("memcache+tls://<cluster_endpoint>:11211?
verify_mode=none").unwrap();

// set a string value
client.set("foo", "bar", 0).unwrap();
```

Connessione tramite il client Memcached Go (Gomemcache)

Consulta <https://github.com/bradfitz/gomemcache>

```
c := New(net.JoinHostPort("{cluster_endpoint}", strconv.Itoa(port)))
c.DialContext = func(ctx context.Context, network, addr string) (net.Conn, error) {
var td tls.Dialer
td.Config = &tls.Config{}
return td.DialContext(ctx, network, addr)
}
foo := &Item{Key: "foo", Value: []byte("fooval"), Flags: 123}
err := c.Set(foo)
```

Connessione tramite il client Memcached Ruby (Dalli)

Consulta <https://github.com/petergoldstein/dalli>

```
require 'dalli'
ssl_context = OpenSSL::SSL::SSLContext.new
ssl_context.ssl_version = :SSLv23
ssl_context.verify_hostname = true
ssl_context.verify_mode = OpenSSL::SSL::VERIFY_PEER
client = Dalli::Client.new("<cluster_endpoint>:11211", :ssl_context => ssl_context);
client.get("abc")
```

Connessione tramite il client Memcached .NET (EnyimMemcachedCore)

Consulta <https://github.com/cnblogs/EnyimMemcachedCore>

```
"MemcachedClient": {  
  "Servers": [  
    {  
      "Address": "{cluster_endpoint}",  
      "Port": 11211  
    }  
  ],  
  "UseSslStream": true  
}
```

Ora puoi procedere alla [Fase 3: \(facoltativo\) pulizia](#).

Fase 3: (facoltativo) pulizia

Utilizzando la AWS Management Console

La procedura seguente elimina una sola cache dall'implementazione. Per eliminare più cache, ripeti la procedura per ogni cache da eliminare. Non occorre attendere la fine dell'eliminazione di una cache prima di avviare la procedura per eliminarne un'altra.

Per eliminare una cache

1. Accedi alla AWS Management Console e apri la console di Amazon ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nel pannello di controllo della console ElastiCache, scegli il motore in esecuzione sulla cache da eliminare. Viene visualizzato un elenco di tutte le cache che eseguono quel motore.
3. Per scegliere la cache da eliminare, seleziona il nome della cache nell'elenco.

Important

Puoi eliminare una sola cache alla volta dalla console ElastiCache. La selezione di più cache disabilita l'operazione di eliminazione.

4. In Actions (Azioni), scegliere Delete (Elimina).
5. Nella schermata di conferma Elimina cache scegli Elimina per eliminare la cache o scegli Annulla per mantenerla.
6. Se si sceglie Elimina, lo stato della cache diventa Eliminazione in corso.

Non appena la cache passa allo stato ELIMINAZIONE IN CORSO, non vengono più addebitati i costi.

Utilizzando la AWS CLI

Il seguente codice elimina la cache my-cache.

```
aws elasticache delete-serverless-cache --serverless-cache-name my-cache
```

L'azione della CLI delete-serverless-cache elimina solo una cache serverless. Per eliminare più cache, chiama delete-serverless-cache per ogni cache serverless che desideri eliminare. Non occorre attendere la fine dell'eliminazione di una cache serverless prima di eliminarne un'altra.

Per Linux, macOS o Unix:

```
aws elasticache delete-serverless-cache \  
  --serverless-cache-name my-cache
```

Per Windows:

```
aws elasticache delete-serverless-cache ^  
  --serverless-cache-name my-cache
```

Per ulteriori informazioni, consulta l'argomento `delete-serverless-cache` della AWS CLI per ElastiCache.

Ora puoi procedere alla [Fasi successive](#).

Fasi successive

Per ulteriori informazioni su ElastiCache, consulta:

- [Lavorare con ElastiCache](#)
- [Scalabilità ElastiCache per Memcached](#)
- [Quote ElastiCache](#)
- [Best practice e strategie di caching ElastiCache](#)
- [Visualizzazione di eventi ElastiCache C](#)

Tutorial: Configurazione di una funzione Lambda per l'accesso ad Amazon ElastiCache in un Amazon VPC

In questo tutorial, esegui quanto indicato di seguito:

- Crea una cache Amazon ElastiCache nell'Amazon Virtual Private Cloud (Amazon VPC) predefinito nella regione us-east-1.
- Crea una funzione Lambda per accedere alla cache ElastiCache. Quando si crea la funzione Lambda, si forniscono gli ID della sottorete nell'Amazon VPC e un gruppo di sicurezza VPC per consentire alla funzione Lambda di accedere alle risorse nel VPC. A scopo illustrativo, in questo tutorial la funzione Lambda genera un valore UUID, lo scrive nella cache e successivamente lo recupera.
- Richiama la funzione Lambda manualmente e verifica che sia in grado di accedere alla cache ElastiCache nel VPC.

Important

In questo tutorial viene utilizzato Amazon VPC predefinito nella regione us-east-1 dell'account in uso. Per ulteriori informazioni su Amazon VPC, consulta [Nozioni di base su Amazon VPC](#) nella Guida per l'utente di Amazon VPC.

Argomenti

- [Fase 1: creazione di una cache ElastiCache](#)
- [Passaggio 2: creazione di una funzione Lambda](#)
- [Fase 3: esecuzione del test della funzione Lambda](#)

Nozioni di base

[Fase 1: creazione di una cache ElastiCache](#)

Fase 1: creazione di una cache ElastiCache

In questa fase viene creata una cache Amazon ElastiCache nell'Amazon Virtual Private Cloud predefinito nella regione us-east-1 dell'account in uso utilizzando l'interfaccia della linea di comando

AWS. Per informazioni sulla creazione di cache serverless ElastiCache utilizzando la console o l'API ElastiCache, consulta [Creazione di un cluster](#) nella Guida per l'utente di ElastiCache per Memcached.

AWS Management Console

Esegui il comando della AWS CLI seguente per creare una nuova cache serverless del cluster Memcached nel VPC predefinito nella regione us-east-1.

Linux

```
aws elasticache create-serverless-cache \  
--serverless-cache-name serverlessCacheForLambda \  
--region us-east-1 \  
--engine memcached
```

Windows

```
aws elasticache create-serverless-cache ^  
--serverless-cache-name serverlessCacheForLambda ^  
--region us-east-1 ^  
--engine memcached
```

Il valore del campo Stato è impostato su CREATING. Possono essere necessari alcuni minuti prima che ElastiCache completi la creazione del cluster.

Per verificare se ElastiCache ha terminato la creazione della cache, utilizza il comando `describe-serverless-caches`.

Linux

```
aws elasticache describe-serverless-caches \  
--serverless-cache-name serverlessCacheforLambda \  
--region us-east-1
```

Windows

```
aws elasticache describe-serverless-caches ^  
--serverless-cache-name serverlessCacheforLambda ^  
--region us-east-1
```

Copia l'indirizzo dell'endpoint presente nell'output. Avrai bisogno di questo indirizzo quando crei il pacchetto di implementazione per la funzione Lambda.

Dopo aver creato la nuova cache, procedi alla [Passaggio 2: creazione di una funzione Lambda](#).

Fase successiva:

[Passaggio 2: creazione di una funzione Lambda](#)

Passaggio 2: creazione di una funzione Lambda

In questa fase si effettuano le operazioni seguenti:

1. Creazione di un pacchetto di implementazione della funzione Lambda utilizzando il codice di esempio fornito.
2. Creazione di un ruolo IAM (ruolo di esecuzione). Quando si carica il pacchetto di implementazione, è necessario specificare questo ruolo in modo che Lambda possa assumerlo per eseguire la funzione per conto dell'utente. La policy di autorizzazione concede ad AWS Lambda le autorizzazioni per configurare le interfacce di rete elastiche o ENI per permettere alla funzione Lambda di accedere alle risorse nel VPC. In questo esempio, la funzione Lambda accede a un cluster ElastiCache nel VPC.
3. Creazione della funzione Lambda caricando il pacchetto di implementazione.

Fase successiva

[Fase 2.1: creazione del pacchetto di implementazione](#)

Fase 2.1: creazione del pacchetto di implementazione

Attualmente il codice di esempio per la funzione Lambda è fornito solo in Python.

Python

Il codice Python di esempio seguente legge e scrive una voce nel cluster ElastiCache. Copia il codice e salvalo in un file denominato `app.py`. Assicurati di sostituire il valore `elasticache_config_endpoint` nel codice con l'indirizzo dell'endpoint che hai copiato alla fase 1.

```
import uuid
import ssl
```

```
from pymemcache.client.base import Client

elasticache_config_endpoint = "serverlesscacheforlambda-
ces85m.serverless.use1.cache.amazonaws.com"
target_port = 11211

context = ssl.create_default_context()

memcached_client = Client((elasticache_config_endpoint, target_port),
    tls_context=context)

def lambda_handler(event, context):

    # create a random UUID - this will be the sample element we add to the cache
    uuid_in = uuid.uuid4().hex

    # put the UUID to the cache
    memcached_client.set("uuid", uuid_in, expire=500, noreply=False)

    # get the item (UUID) from the cache
    result = memcached_client.get("uuid")
    decoded_result = result.decode("utf-8")

    # check the retrieved item matches the item added to the cache and print
    # the results
    if decoded_result == uuid_in:
        print(f"Success: Inserted {uuid_in}. Fetched {decoded_result} from Memcached.")
    else:
        raise Exception(f"Bad value retrieved. Expected {uuid_in}, got
{decoded_result}")

    return "Fetched value from Memcached"
```

Questo codice utilizza la libreria [pymemcache](#) Python per inserire elementi nella cache e recuperarli. Per creare un pacchetto di implementazione contenente pymemcache, attieniti alla procedura descritta di seguito.

1. Nella directory del progetto contenente il file del codice sorgente `app.py`, crea una cartella package in cui installare la libreria pymemcache.

```
mkdir package
```

2. Installa pymemcache utilizzando pip.

```
pip install --target ./package pymemcache
```

3. Crea un file .zip contenente la libreria pymemcache. Su Linux e macOS, esegui il comando seguente. Su Windows, utilizza il tuo strumento di compressione preferito per creare un file .zip con la libreria pymemache nella directory principale.

```
cd package
zip -r ../my_deployment_package.zip .
```

4. Aggiungi il codice della funzione al file .zip. Su Linux e macOS, esegui il comando seguente. Su Windows, utilizza il tuo strumento di compressione preferito per aggiungere app.py nella directory principale del file .zip.

```
cd ..
zip my_deployment_package.zip app.py
```

Fase successiva

[Fase 2.2: creazione del ruolo IAM \(ruolo di esecuzione\)](#)

Fase 2.2: creazione del ruolo IAM (ruolo di esecuzione)

In questa fase si crea un ruolo AWS Identity and Access Management (IAM) utilizzando il tipo di ruolo e la policy di accesso predefiniti riportati di seguito.

- Ruolo di servizio AWS di tipo AWS Lambda - Questo ruolo concede ad AWS Lambda le autorizzazioni per assumere il ruolo.
- AWSLambdaVPCLambdaAccessExecutionRole - Questa è la policy di autorizzazione di accesso che si collega al ruolo, che concede l'autorizzazione per le azioni EC2 richieste da AWS Lambda per gestire le interfacce di rete elastiche (ENI). Puoi visualizzare questa policy gestita da AWS nella console IAM.

Per ulteriori informazioni sui ruoli utente di IAM, consulta [Ruoli \(Delega e Federazione\)](#) nella Guida per l'utente IAM.

Per creare il ruolo IAM, utilizzare la procedura seguente.

Per creare un ruolo IAM (esecuzione)

1. Registrati alla Console di gestione di AWS e apri la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.
2. Scegli Roles (Ruoli), quindi Create role (Crea ruolo).
 - In Tipo di entità attendibile scegli Servizio AWS, quindi seleziona Lambda in Casi d'uso. Questa operazione concede al servizio AWS Lambda le autorizzazioni per assumere il ruolo. Seleziona Successivo.
 - In Aggiungi autorizzazioni cerca **AWSLambdaVPCAccessExecutionRole** e seleziona la casella di controllo accanto alla policy.
 - Seleziona Successivo.
 - In Role Name (Nome ruolo), utilizza un nome univoco nell'account AWS (ad esempio, lambda-vpc-execution-role).
 - Scegli Crea ruolo.
3. Copiare il ruolo ARN. Sarà necessario per la creazione della funzione Lambda nella fase successiva.

Fase successiva

[Fase 2.3: caricamento del pacchetto di implementazione \(creazione della funzione Lambda\)](#)

Fase 2.3: caricamento del pacchetto di implementazione (creazione della funzione Lambda)

In questa fase viene creata la funzione Lambda (AccessMemcached) mediante il comando AWS CLI `create-function`.

Dalla directory del progetto contenente il file.zip del pacchetto di implementazione, esegui il seguente comando `create-function` della CLI Lambda.

Per l'opzione `role` utilizza l'ARN del ruolo di esecuzione creato alla fase 2.2. Per `vpc-config` immetti gli elenchi separati da virgole delle sottoreti del VPC predefinito e l'ID del gruppo di sicurezza del VPC predefinito. Questi valori sono disponibili nella [console Amazon VPC](#). Per trovare le sottoreti del VPC predefinito, scegli I tuoi VPC, quindi seleziona il VPC predefinito dell'Account AWS. Per trovare il gruppo di sicurezza per il VPC, scegli Gruppi di sicurezza in Sicurezza. Assicurati di aver selezionato la regione us-east-1.

Per Linux, macOS o Unix:

```
aws lambda create-function \  
--function-name AccessMemcached \  
--region us-east-1 \  
--zip-file fileb://my_deployment_package.zip \  
--role arn:aws:iam::123456789012:role/lambda-vpc-execution-role \  
--handler app.lambda_handler \  
--runtime python3.11 \  
--timeout 30 \  
--vpc-config SubnetIds=comma-separated-vpc-subnet-ids,SecurityGroupIds=default-security-group-id \  

```

Per Windows:

```
aws lambda create-function ^  
--function-name AccessMemcached ^  
--region us-east-1 ^  
--zip-file fileb://path-to/my_deployment_package.zip ^  
--role arn:aws:iam::123456789012:role/lambda-vpc-execution-role ^  
--handler app.lambda_handler ^  
--runtime python3.11 ^  
--timeout 30 ^  
--vpc-config SubnetIds=comma-separated-vpc-subnet-ids,SecurityGroupIds=default-security-group-id ^  

```

Facoltativamente, è possibile caricare il file .zip in un bucket Amazon S3 nella stessa regione AWS e quindi specificare il nome del bucket e dell'oggetto nel comando precedente. È necessario sostituire il parametro `--zip-file` con il parametro `--code`, come illustrato di seguito:

```
--code S3Bucket=bucket-name,S3Key=zip-file-object-key
```

È anche possibile creare la funzione Lambda mediante la console AWS Lambda. Durante la creazione della funzione, scegliere un VPC per Lambda e quindi selezionare le sottoreti e i gruppi di sicurezza nei campi forniti.

Fase successiva

[Fase 3: esecuzione del test della funzione Lambda](#)

Fase 3: esecuzione del test della funzione Lambda

In questa fase si invoca la funzione Lambda manualmente utilizzando il comando `invoke`. Durante l'esecuzione, la funzione Lambda genera un valore UUID e lo scrive nel cluster ElastiCache specificato nel codice Lambda. Successivamente la funzione Lambda recupera la voce dalla cache.

1. Invocare la funzione Lambda (`AccessMemCache`) mediante il comando `invoke` di AWS Lambda.

Per Linux, macOS o Unix:

```
aws lambda invoke \  
--function-name AccessMemCache \  
--region us-east-1 \  
output.txt
```

Per Windows:

```
aws lambda invoke ^  
--function-name AccessMemCache ^  
--region us-east-1 ^  
output.txt
```

2. Eseguire le operazioni seguenti per verificare che la funzione Lambda sia stata eseguita nel modo corretto:
 - Esaminare il file `output.txt`.
 - Verifica i risultati in File di log CloudWatch aprendo la console [CloudWatch](#) e scegliendo il gruppo di log per la funzione (`/aws/lambda/AccessMemcached`). Il flusso di log genera un output simile al seguente:

```
Success: Inserted 05fcf2e4d6c942209acc89ea79b5b15e. Fetched  
05fcf2e4d6c942209acc89ea79b5b15e from Memcached.
```

- Esaminare i risultati nella console AWS Lambda.

Esercitazioni e video relativi a ElastiCache

I tutorial seguenti illustrano le attività di interesse per l'utente di Amazon ElastiCache.

- [Video relativi a ElastiCache](#)
- [Tutorial: Configurazione di una funzione Lambda per l'accesso a Amazon ElastiCache in un Amazon VPC](#)

Video relativi a ElastiCache

Di seguito sono riportati i video che consentono di apprendere concetti di base e avanzati relativi a Amazon ElastiCache. Per informazioni sulla formazione di AWS, vedere [Formazione e Certificazione di AWS](#).

Argomenti

- [Video introduttivi](#)
- [Video avanzati](#)

Video introduttivi

I video seguenti rappresentano un'introduzione ad Amazon ElastiCache.

Argomenti

- [AWS re:Invent 2020: Cosa c'è di nuovo in Amazon ElastiCache](#)
- [AWS re:Invent 2019: Cosa c'è di nuovo in Amazon ElastiCache](#)
- [AWS re:Invent 2017: Cosa c'è di nuovo in Amazon ElastiCache](#)
- [DAT204—Costruire applicazioni scalabili su AWS Servizi NoSQL \(re:Invent 2015\)](#)
- [DAT207—Accelerare le prestazioni delle applicazioni con Amazon ElastiCache \(AWS re:Invent 2013\)](#)

AWS re:Invent 2020: Cosa c'è di nuovo in Amazon ElastiCache

[AWS re:Invent 2020: Cosa c'è di nuovo in Amazon ElastiCache](#)

AWS re:Invent 2019: Cosa c'è di nuovo in Amazon ElastiCache

[AWS re:Invent 2019: Cosa c'è di nuovo in Amazon ElastiCache](#)

AWS re:Invent 2017: Cosa c'è di nuovo in Amazon ElastiCache

[AWS re:Invent 2017: Cosa c'è di nuovo in Amazon ElastiCache](#)

DAT204—Costruire applicazioni scalabili su AWS Servizi NoSQL (re:Invent 2015)

In questa sessione illustreremo i vantaggi dei database NoSQL e forniremo una panoramica dei principali servizi NoSQL offerti da AWS: Amazon DynamoDB e Amazon ElastiCache. Quindi,

ascolteremo gli esponenti di due principali aziende nostre clienti, Expedia e Mapbox, che ci parleranno dei loro casi d'uso, dei problemi legati all'architettura e del modo in cui hanno risolto tali problemi utilizzando i servizi AWS NoSQL, inclusi modelli di progettazione e best practice. Al termine di questa sessione avrai una conoscenza più approfondita di NoSQL e delle sue potenti funzionalità, sarai pronto ad affrontare con sicurezza le sfide del tuo database.

[DAT204—Costruire applicazioni scalabili su Servizi NoSQL AWS\(re:Invent 2015\)](#)

[DAT207—Accelerare le prestazioni delle applicazioni con Amazon ElastiCache \(AWS re:Invent 2013\)](#)

In questo video, scoprirai come utilizzare Amazon ElastiCache per distribuire facilmente un sistema di cache in memoria per velocizzare le prestazioni dell'applicazione. Ti illustreremo le modalità di utilizzo di Amazon ElastiCache per migliorare la latenza dell'applicazione e ridurre il carico sui server del database. Inoltre, ti mostreremo come costruire un livello di caching facile da gestire e dimensionare man mano che l'applicazione cresce. Durante questa sessione, esamineremo diversi scenari e casi d'uso che potrebbero trarre vantaggio dall'attivazione del caching e discuteremo delle caratteristiche di Amazon ElastiCache.

[DAT207 - Accelerare le prestazioni delle applicazioni con Amazon ElastiCache \(re:Invent 2013\)](#)

Video avanzati

I seguenti video trattano argomenti più avanzati relativi ad Amazon ElastiCache.

Argomenti

- [Progettare per il successo con le migliori pratiche di Amazon ElastiCache \(re:Invent 2020\)](#)
- [Supercarica le tue applicazioni in tempo reale con Amazon ElastiCache \(re:Invent 2019\)](#)
- [Migliori pratiche: migrazione di cluster Redis da Amazon EC2 a ElastiCache \(re:Invent 2019\)](#)
- [Dimensionare una piattaforma di Fantasy Sports con Amazon ElastiCache e Amazon Aurora STP11 \(re:Invent 2018\)](#)
- [Redis affidabile e scalabile nel cloud con Amazon ElastiCache \(re:Invent 2018\)](#)
- [Approfondimento di ElastiCache: Modelli di progettazione per archivi di dati in memoria \(re:Invent 2018\)](#)
- [DAT305—Approfondimento Amazon ElastiCache \(re:Invent 2017\)](#)
- [DAT306—Approfondimento Amazon ElastiCache \(re:Invent 2016\)](#)
- [DAT407 - Approfondimento Amazon ElastiCache \(Re:Invent 2015\)](#)

- [SDD402—Approfondimento Amazon ElastiCache \(re:Invent 2014\)](#)
- [DAT307—Approfondimento dell'architettura e dei modelli di progettazione di Amazon ElastiCache \(re:Invent 2013\)](#)

[Progettare per il successo con le migliori pratiche di Amazon ElastiCache \(re:Invent 2020\)](#)

Con la crescita esplosiva delle applicazioni in tempo reale di importanza primaria per il business basate su Redis, disponibilità, scalabilità e sicurezza sono diventate le principali considerazioni. Scopri le best practice per la configurazione di Amazon ElastiCache per il successo con dimensionamento online, disponibilità elevata nelle distribuzioni Multi-AZ e configurazioni di sicurezza.

[Progettare per il successo con le migliori pratiche di Amazon ElastiCache \(re:Invent 2020\)](#)

[Supercarica le tue applicazioni in tempo reale con Amazon ElastiCache \(re:Invent 2019\)](#)

Con la rapida crescita dell'adozione del cloud e i nuovi scenari che offre, le applicazioni richiedono una latenza di microsecondi e una velocità effettiva elevata per supportare milioni di richieste al secondo. Gli sviluppatori si sono tradizionalmente affidati a hardware specializzati e soluzioni alternative, come database basati su disco combinati con tecniche di riduzione dei dati, per gestire i dati per applicazioni in tempo reale. Questi approcci possono essere costosi e non scalabili. Scopri come migliorare le prestazioni delle applicazioni in tempo reale utilizzando Amazon ElastiCache completamente gestito e in memoria per prestazioni estreme, scalabilità elevata, disponibilità e sicurezza.

[Supercarica le tue applicazioni in tempo reale con Amazon ElastiCache \(re:Invent 2019:\)](#)

[Migliori pratiche: migrazione di cluster Redis da Amazon EC2 a ElastiCache \(re:Invent 2019\)](#)

Gestire i cluster di Redis da soli può essere difficile. È necessario effettuare il provisioning di hardware, software di patch, eseguire il backup dei dati e monitorare costantemente i carichi di lavoro. Con la funzionalità di migrazione online appena rilasciata per Amazon ElastiCache, è ora possibile spostare facilmente i dati da Redis auto-ospitato su Amazon EC2 a Amazon ElastiCache completamente gestito, con la modalità cluster disabilitata. In questa sessione, imparerai a conoscere il nuovo strumento di migrazione online, consultare una demo e, cosa più importante, apprendrai le best practice concrete per una migrazione fluida ad Amazon ElastiCache.

[Migliori pratiche: migrazione di cluster Redis da Amazon EC2 a ElastiCache \(re:Invent 2019\)](#)

[Dimensionare una piattaforma di Fantasy Sports con Amazon ElastiCache e Amazon Aurora STP11 \(re:Invent 2018\)](#)

Dream11 è la startup leader in India nel settore dello sport. Ha una base crescente di oltre 40 milioni di utenti che praticano più sport, tra cui fantasy cricket, calcio e basket, e attualmente serve un milione di utenti simultanei, che producono tre milioni di richieste al minuto sotto un tempo di risposta di 50 millisecondi. In questo discorso, il CTO di Dream11 Amit Sharma spiega come l'azienda utilizza Amazon Aurora e Amazon ElastiCache per gestire il traffico flash, che può triplicare all'interno di una finestra di risposta di 30 secondi. Sharma parla anche di dimensionare le transazioni senza bloccare e condivide i passaggi per gestire il traffico flash, servendo così cinque milioni di utenti attivi giornalmente. Titolo completo: AWS re:Invent 2018: Scalare una piattaforma di Fantasy Sports con Amazon ElastiCache e Amazon Aurora (STP11)

[Dimensionare una piattaforma di Fantasy Sports con Amazon ElastiCache e Amazon Aurora STP11 \(re:Invent 2018\)](#)

[Redis affidabile e scalabile nel cloud con Amazon ElastiCache \(re:Invent 2018\)](#)

Questa sessione tratta le funzionalità e i miglioramenti del nostro servizio compatibile con Redis, Amazon ElastiCache for Redis. Trattiamo funzioni chiave, come Redis 5, scalabilità e miglioramenti delle prestazioni, sicurezza e conformità e molto altro ancora. Discutiamo anche le funzionalità future e i casi di studio dei clienti.

[Redis affidabile e scalabile nel cloud con Amazon ElastiCache \(re:Invent 2018\)](#)

[Approfondimento di ElastiCache: Modelli di progettazione per archivi di dati in memoria \(re:Invent 2018\)](#)

In questa sessione, forniamo uno sguardo dietro le quinte per conoscere la progettazione e l'architettura di Amazon ElastiCache. Vedi i modelli di progettazione comuni con le nostre offerte Redis e Memcached e scopri come i clienti li utilizzano per l'elaborazione dei dati in memoria per ridurre la latenza e migliorare la velocità effettiva delle applicazioni. Rivedremo le best practice, i modelli e gli anti-modelli di progettazione di ElastiCache.

[Approfondimento di ElastiCache: Modelli di progettazione per archivi di dati in memoria \(re:Invent 2018\)](#)

[DAT305—Approfondimento Amazon ElastiCache \(re:Invent 2017\)](#)

Guarda dietro le quinte per scoprire la progettazione e l'architettura di Amazon ElastiCache. Vedi i modelli di progettazione comuni con le nostre offerte Memcached e Redis e scopri come i clienti

li hanno utilizzati per le operazioni in-memory per ridurre la latenza e migliorare il throughput delle applicazioni. Nel corso di questo video, rivedremo le best practice, i modelli e gli anti-modelli di progettazione di ElastiCache.

Il video introduce quanto segue:

- ElastiCache per il resharding online di Redis
- Sicurezza e crittografia in ElastiCache
- ElastiCache for Redis versione 3.2.10

[DAT305—Approfondimento Amazon ElastiCache \(re:Invent 2017\)](#)

DAT306—Approfondimento Amazon ElastiCache (re:Invent 2016)

Guarda dietro le quinte per scoprire la progettazione e l'architettura di Amazon ElastiCache. Vedi i modelli di progettazione comuni con le nostre offerte Memcached e Redis e scopri come i clienti li hanno utilizzati per le operazioni in-memory per ridurre la latenza e migliorare il throughput delle applicazioni. Nel corso di questa sessione, rivedremo le best practice, i modelli e gli anti-modelli di progettazione correlati di ElastiCache.

[DAT306—Approfondimento Amazon ElastiCache \(re:Invent 2016\)](#)

DAT407 - Approfondimento Amazon ElastiCache (Re:Invent 2015)

Guarda dietro le quinte per scoprire la progettazione e l'architettura di Amazon ElastiCache. Vedi i modelli di progettazione comuni delle nostre offerte Memcached e Redis e scopri come i clienti li hanno utilizzati per le operazioni in-memory e per migliorare la latenza potenziata e il throughput delle applicazioni. Nel corso di questa sessione, rivedremo le best practice, i modelli e gli anti-modelli di progettazione correlati di Amazon ElastiCache.

[DAT407—Approfondimento Amazon ElastiCache \(re:Invent 2015\)](#)

SDD402—Approfondimento Amazon ElastiCache (re:Invent 2014)

In questo video esamineremo i comuni casi di utilizzo del caching, i motori Memcached e Redis, i modelli che ti consentono di determinare quale modello è migliore per le tue esigenze, l'hashing coerente e molto altro. Tutti elementi che consentono di creare applicazioni veloci e scalabili. Frank Wiebe, principale scienziato in Adobe, illustra come Adobe utilizza Amazon ElastiCache per migliorare l'esperienza cliente e dimensionare il business.

[DAT402—Approfondimento Amazon ElastiCache \(re:Invent 2014\)](#)

DAT307—Approfondimento dell'architettura e dei modelli di progettazione di Amazon ElastiCache (re:Invent 2013)

In questo video esamineremo caching, strategie di caching, dimensionamento e monitoraggio. Confronteremo, inoltre, i motori Memcached e Redis. Nel corso di questa sessione, rivedremo anche le best practice e i modelli di progettazione correlati ad Amazon ElastiCache.

[DAT307 - Approfondimento dell'architettura e dei modelli di progettazione di Amazon ElastiCache \(AWS re:Invent 2013\)](#).

Scelta di regioni e zone di disponibilità

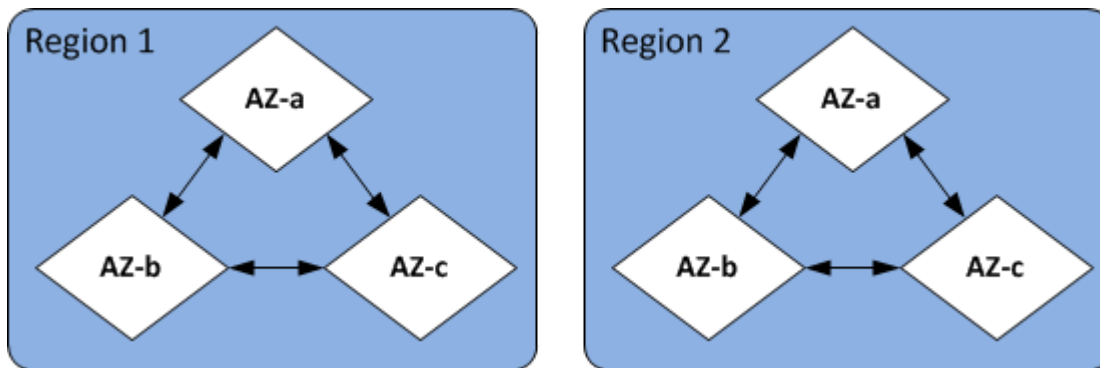
AWS Le risorse di cloud computing sono ospitate in strutture di data center ad alta disponibilità. Per fornire ulteriore scalabilità e affidabilità, queste strutture di data center sono in ubicazioni fisiche diverse. Tali località sono categorizzate in base a regioni e zone di disponibilità.

AWS Le regioni sono ampie e ampiamente distribuite in località geografiche separate. Le zone di disponibilità sono località distinte all'interno di una AWS regione progettate per essere isolate dai guasti in altre zone di disponibilità. Forniscono connettività di rete economica e a bassa latenza ad altre zone di disponibilità nella stessa regione. AWS

⚠ Important

Ciascuna regione è completamente indipendente. Qualsiasi ElastiCache attività avviata (ad esempio, la creazione di cluster) viene eseguita solo nella regione predefinita corrente.

Per creare o utilizzare un cluster in una regione specifica, utilizza l'endpoint del servizio regionale corrispondente. Per gli endpoint del servizio, consulta [Regioni ed endpoint supportati](#).



Regioni e zone di disponibilità

Argomenti

- [Considerazioni sulle zone di disponibilità](#)
- [Regioni ed endpoint supportati](#)
- [Ubicazione dei nodi](#)
- [Utilizzo di local zones con ElastiCache](#)
- [Utilizzo Outposts](#)

Considerazioni sulle zone di disponibilità

La distribuzione dei nodi Memcached su più zone di disponibilità all'interno di un'area consente di proteggere l'utente dall'impatto di un guasto catastrofico, ad esempio una perdita di alimentazione all'interno di una zona di disponibilità.

Cache serverless

ElastiCache la memorizzazione nella cache senza server crea una cache ad alta disponibilità che si estende su più zone di disponibilità. Puoi specificare sottoreti da diverse zone di disponibilità e dallo stesso VPC mentre crei la cache o ElastiCache sceglierai automaticamente le sottoreti dal tuo VPC predefinito.

Progettarne uno personalizzato per il cluster Memcached ElastiCache


Un cluster Memcached può avere fino a 300 nodi. Quando si creano o si aggiungono nodi al cluster Memcached, è possibile specificare una singola zona di disponibilità per tutti i nodi, consentire di ElastiCache scegliere una singola zona di disponibilità per tutti i nodi, specificare le zone di disponibilità per ogni nodo o consentire di scegliere una zona di disponibilità ElastiCache per ogni nodo. È possibile creare nuovi nodi in diverse zone di disponibilità quando vengono aggiunti a un cluster Memcached esistente. Una volta creato un nodo di cache, la relativa zona di disponibilità non può essere modificata.

Se desideri che un cluster in un singolo cluster di zona di disponibilità abbia i suoi nodi distribuiti su più zone di disponibilità, ElastiCache puoi creare nuovi nodi nelle varie zone di disponibilità. È quindi possibile eliminare alcuni o tutti i nodi della cache originali. Consigliamo questo approccio.

Per eseguire la migrazione dei nodi Memcached da una singola zona di disponibilità a più zone di disponibilità

1. Modificare il cluster creando nuovi nodi della cache nelle zone di disponibilità in cui si desidera. Nella richiesta, procedere come segue:
 - Imposta `AZMode`(CLI: `--az-mode`) `across-az`.
 - Imposta `NumCacheNodes`(CLI: `--num-cache-nodes`) al numero di nodi della cache attualmente attivi più il numero di nuovi nodi della cache che si desidera creare.
 - Imposta `NewAvailabilityZones`(CLI: `--new-availability-zones`) in un elenco delle zone in cui si desidera creare i nuovi nodi della cache. Per consentire di ElastiCache determinare la zona di disponibilità per ogni nuovo nodo, non specificare un elenco.

- Imposta `ApplyImmediately`(CLI:`--apply-immediately`) su `VERO`.

 Note

Se non si utilizza l'individuazione automatica, assicurarsi di aggiornare l'applicazione client con i nuovi endpoint del nodo di cache.

Prima di passare al passaggio successivo, assicurarsi che i nodi Memcached siano completamente creati e disponibili.

2. Modificare il cluster rimuovendo i nodi che non si desidera più nella zona di disponibilità originale. Nella richiesta, procedere come segue:

- Imposta `NumCacheNodes` (CLI:`--num-cache-nodes`) al numero di nodi di cache attivi desiderati dopo l'applicazione della modifica.
- Imposta `CacheNodeIdsToRemove` (CLI:`--nodes-to-remove`) in un elenco dei nodi di cache che si desidera rimuovere dal cluster.

Il numero di ID di nodi di cache elencati deve essere uguale al numero di nodi attivi meno il valore in `NumCacheNodes`.

- (Facoltativo) Impostare `ApplyImmediately` (CLI:`--apply-immediately`) su `VERO`.

Se non imposti `ApplyImmediately` (CLI:`--apply-immediately`) su `VERO`, le eliminazioni dei nodi avverranno nella prossima finestra di manutenzione.

Regioni ed endpoint supportati

Amazon ElastiCache è disponibile in più AWS regioni. Ciò significa che puoi avviare ElastiCache cluster in località che soddisfano i tuoi requisiti. Ad esempio, puoi lanciare il prodotto nella AWS regione più vicina ai tuoi clienti o farlo in una AWS regione particolare per soddisfare determinati requisiti legali.

Ogni regione è pensata per essere completamente isolata dalle altre regioni. All'interno di ciascuna regione ci sono più zone di disponibilità. ElastiCache Le cache serverless replicano automaticamente i dati su più zone di disponibilità (ad eccezione us-west-1 dei casi in cui i dati vengono replicati in due zone di disponibilità) per un'elevata disponibilità. Quando si progetta il proprio ElastiCache cluster, è possibile scegliere di avviare i nodi in diverse AZ per ottenere la tolleranza agli errori. Per ulteriori informazioni sulle regioni e sulle zone di disponibilità, consulta [Scelta di regioni e zone di disponibilità](#) all'inizio di questo argomento.

Regioni in cui ElastiCache è supportato

Nome regione/Regione	Endpoint	Protocollo
Stati Uniti orientali (Ohio) us-east-2	elasticache.us-east-2.amazonaws.com	HTTPS
Regione Stati Uniti orientali (Virginia settentrionale) us-east-1	elasticache.us-east-1.amazonaws.com	HTTPS
Regione Stati Uniti occidentali (California settentrionale) us-west-1	elasticache.us-west-1.amazonaws.com	HTTPS
Stati Uniti occidentali (Oregon)	elasticache.us-west-2.amazonaws.com	HTTPS

Nome regione/Regione	Endpoint	Protocollo	
us-west-2			
Regione Canada (Centrale) ca-central-1	elasticache.ca-central-1.amazonaws.com	HTTPS	
Regione Canada (Occidentale) ca-west-1	elasticache.ca-west-1.amazonaws.com	HTTPS	
Asia Pacifico (Giacarta) ap-southeast-3	elasticache.ap-southeast-3.amazonaws.com	HTTPS	
Regione Asia Pacifico (Mumbai) ap-south-1	elasticache.ap-south-1.amazonaws.com	HTTPS	
Regione Asia Pacifico (Hyderabad) ap-south-2	elasticache.ap-south-2.amazonaws.com	HTTPS	
Regione Asia Pacifico (Tokyo) ap-northeast-1	elasticache.ap-northeast-1.amazonaws.com	HTTPS	
Regione Asia Pacifico (Seoul) ap-northeast-2	elasticache.ap-northeast-2.amazonaws.com	HTTPS	

Nome regione/Regione	Endpoint	Protocollo	
Regione Asia Pacifico (Osaka-Locale) ap-northeast-3	elasticache.ap-northeast-3.amazonaws.com	HTTPS	
Regione Asia Pacifico (Singapore) ap-southeast-1	elasticache.ap-southeast-1.amazonaws.com	HTTPS	
Asia Pacifico (Sydney) ap-southeast-2	elasticache.ap-southeast-2.amazonaws.com	HTTPS	
Regione Europa (Francoforte) eu-central-1	elasticache.eu-central-1.amazonaws.com	HTTPS	
Regione Europa (Zurigo) eu-central-2	elasticache.eu-central-2.amazonaws.com	HTTPS	
Regione Europa (Stoccolma) eu-north-1	elasticache.eu-north-1.amazonaws.com	HTTPS	
Regione Medio Oriente (Bahrein) me-south-1	elasticache.me-south-1.amazonaws.com	HTTPS	

Nome regione/Regione	Endpoint	Protocollo	
Regione Medio Oriente (Emirati Arabi Uniti) me-central-1	elasticache.me-central-1.amazonaws.com	HTTPS	
Europa (Irlanda) eu-west-1	elasticache.eu-west-1.amazonaws.com	HTTPS	
Regione Europa (Londra) eu-west-2	elasticache.eu-west-2.amazonaws.com	HTTPS	
Regione UE (Parigi) eu-west-3	elasticache.eu-west-3.amazonaws.com	HTTPS	
Regione Europa (Milano) eu-south-1	elasticache.eu-south-1.amazonaws.com	HTTPS	
Regione Europa (Spagna) eu-south-2	elasticache.eu-south-2.amazonaws.com	HTTPS	
Regione Sud America (San Paolo) sa-east-1	elasticache.sa-east-1.amazonaws.com	HTTPS	

Nome regione/Regione	Endpoint	Protocollo
Regione Cina (Pechino) cn-north-1	elasticache.cn-north-1.amazonaws.com.cn	HTTPS
Regione Cina (Ningxia) cn-northwest-1	elasticache.cn-northwest-1.amazonaws.com.cn	HTTPS
Regione Asia Pacifico (Hong Kong) ap-east-1	elasticache.ap-east-1.amazonaws.com	HTTPS
Regione Africa (Città del Capo) af-south-1	elasticache.af-south-1.amazonaws.com	HTTPS
Regione di Israele (Tel Aviv) il-central-1	elasticache.il-central-1.amazonaws.com	HTTPS
AWS GovCloud (Stati Uniti occidentali) us-gov-west-1	elasticache.us-gov-west-1.amazonaws.com	HTTPS
AWS GovCloud (Stati Uniti orientali) us-gov-east-1	elasticache.us-gov-east-1.amazonaws.com	HTTPS

Per informazioni sull'utilizzo di AWS GovCloud (Stati Uniti) con ElastiCache, vedi [Servizi nella regione AWS GovCloud \(Stati Uniti\): ElastiCache](#).

Alcune regioni supportano un set secondario di tipi di nodo. Per una tabella dei tipi di nodi supportati per AWS regione, vedere [Tipi di nodi supportati in base alla regione AWS](#).

Per una tabella di AWS prodotti e servizi per regione, vedi [Prodotti e servizi per regione](#).

Ubicazione dei nodi

Amazon ElastiCache supporta la localizzazione di tutti i nodi di un cluster in una o più zone di disponibilità (AZ). Inoltre, se scegli di localizzare i tuoi nodi in più AZ (scelta consigliata), ti ElastiCache consente di scegliere le AZ per ogni nodo o di scegliere quelle ElastiCache per te.

Ubicando i nodi in diverse zone di disponibilità, elimini la possibilità che un errore, ad esempio un'interruzione di corrente, in una zona di disponibilità causi l'interruzione dell'intero sistema.

Puoi specificare una zona di disponibilità per ciascun nodo quando crei un cluster o aggiungendo nodi quando modifichi un cluster esistente. Per ulteriori informazioni, consulta gli argomenti seguenti:

- [Creazione di un cluster](#)
- [Modifica di un cluster ElastiCache](#)
- [Aggiunta di nodi a un cluster](#)

Utilizzo di local zones con ElastiCache

Una zona locale è un'estensione di una regione AWS che è geograficamente vicina agli utenti. Puoi estendere qualsiasi servizio virtual private cloud (VPC) dalla regione AWS principale a Local Zones tramite la creazione di una nuova sottorete da assegnare alla zona locale. Quando si crea una sottorete in una zona locale, il VPC viene esteso anche a tale zona locale. La sottorete nell'area locale funziona allo stesso modo delle altre sottoreti nel VPC.

Utilizzando Local Zones è possibile posizionare risorse come un cluster ElastiCache in più posizioni vicine agli utenti.

Quando si crea un cluster ElastiCache, è possibile scegliere una sottorete in una zona locale. Le Local Zones hanno le loro connessioni a Internet e supportano AWS Direct Connect. Pertanto, le risorse create in un'area locale possono servire gli utenti locali con comunicazioni a latenza molto bassa. Per ulteriori informazioni, consulta [AWS Local Zones](#).

Una zona locale è rappresentata da un codice regione AWS seguito da un identificatore che indica la posizione, ad esempio `us-west-2-lax-1a`.

Al momento, le Local Zones disponibili sono `us-west-2-lax-1a` e `us-west-2-lax-1b`.

Le seguenti limitazioni si applicano a ElastiCache per Local Zones:

- I seguenti tipi di nodi sono supportati dalle Local Zones:

- Generazione attuale:

Tipi di nodi M5: `cache.m5.large`, `cache.m5.xlarge`, `cache.m5.2xlarge`, `cache.m5.4xlarge`, `cache.m5.12xlarge`, `cache.m5.24xlarge`

Tipi di nodi R5: `cache.r5.large`, `cache.r5.xlarge`, `cache.r5.2xlarge`, `cache.r5.4xlarge`, `cache.r5.12xlarge`, `cache.r5.24xlarge`

Tipi di nodo T3: `cache.t3.micro`, `cache.t3.small`, `cache.t3.medium`

Abilitazione di una zona locale

1. Attivare la zona locale nella console di Amazon EC2.

Per ulteriori informazioni, vedere [Abilitazione di Local Zones](#) nella Guida per l'utente di Amazon EC2 per le istanze Linux.

2. Creare una sottorete nella zona locale.

Per ulteriori informazioni, consulta [Creazione di una sottorete nel VPC](#) nella Guida per l'utente di Amazon VPC.

3. Creare un gruppo di sottoreti di ElastiCache in Local Zone.

Quando si crea un gruppo di sottoreti di ElastiCache, scegliere il gruppo Zona di disponibilità per la zona locale.

Per ulteriori informazioni, consulta [Creazione di un gruppo di sottoreti](#) nella guida per l'utente ElastiCache.

4. Creare un cluster ElastiCache for MemCached che utilizza la sottorete ElasticaCache nella zona locale.

Per ulteriori informazioni, consulta [Creazione di un cluster Memcached \(console\)](#).

Utilizzo Outposts

AWS Outposts è un servizio completamente gestito che estende l'infrastruttura AWS, i servizi, le API e gli strumenti alla sede del cliente. Fornendo l'accesso locale all'infrastruttura gestita AWS, AWSOutposts consente ai clienti di creare ed eseguire applicazioni in locale utilizzando le stesse

interfacce di programmazione delle regioni AWS, utilizzando al contempo le risorse di calcolo e archiviazione locali per le esigenze di bassa latenza e minore elaborazione dei dati locali. Un Outpost è un pool di capacità di AWS elaborazione e storage distribuito presso la sede del cliente. AWS gestisce, monitora e gestisce questa capacità come parte di una regione AWS. Puoi creare sottoreti nell'Outpost e specificarle quando crei risorse quali cluster ElastiCache.

Note

In questa versione si applicano le seguenti limitazioni:

- ElastiCache per Outposts supporta solo le famiglie di nodi M5 e R5.
- Multi-AZ (la replica cross-outpost non è supportata).
- ElastiCache per gli Outposts non è supportato nelle seguenti aree: cn-northwest-1, cn-northwest-1 e ap-northeast-3.

Utilizzo degli Outposts con la console Memcached

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Scegliere Memcached nel pannello di navigazione.
3. Nello stato Motore cluster, selezionare Memcached.
4. In Posizione, selezionare On-premises: crea le tue istanze ElastiCache su AWS Outposts.

Configurare le opzioni locali

È possibile scegliere un Outpost disponibile per aggiungere il cluster di cache oppure, se non ci sono Outposts disponibili, crearne uno nuovo seguendo la procedura seguente:

In Opzioni locali:

1. In Impostazioni Memcached:
 - a. Name (Nome), inserire un nome per il cluster Memcached.
 - b. Descrizione cluster: inserire una descrizione per il cluster Memcached.
 - c. Compatibilità della versione del motore.: la versione del motore è basata sulla AWS regione dell'Outpost

- d. Port (Porta), accettare la porta predefinita, 11211. Se esiste un valido motivo per utilizzare una porta diversa, digitare il numero della porta.
 - e. Gruppo di parametri: Utilizzare il menu a discesa per scegliere un gruppo di parametri di default o personalizzato.
 - f. Tipo di nodo: le istanze disponibili si basano sulla disponibilità degli Outposts. Dall'elenco a discesa, scegliere Outposts e quindi scegliere un tipo di nodo disponibile che si desidera utilizzare per questo cluster. Quindi, scegliere Salva.
 - g. Numero di nodi: immettere il numero di nodi desiderati nel cluster.
2. In Impostazioni avanzate Memcached:
- a. Gruppo di sottoreti: Dall'elenco, seleziona Crea nuova.
 - Nome: Inserire un nome breve per il gruppo di sottoreti.
 - Descrizione: Inserire una descrizione del gruppo di sottoreti.
 - ID VPC: l'ID VPC deve corrispondere al VPC dell'Avamposto.
 - Zona di disponibilità o Outpost: seleziona l'Outpost che stai utilizzando.
 - ID sottorete: scegliere un ID di sottorete disponibile per l'Avamposto. Se non sono disponibili ID di sottoreti, devi crearli. Per ulteriori dettagli, vedi [Crea una sottorete](#).
 - b. Selezionare Crea.

Visualizzazione dei dettagli del cluster Outpost

Nella pagina elenco Memcached scegliere un cluster appartenente a un Outpost AWS e annotare quanto segue quando si visualizzano i dettagli del cluster:

- Availability zone (Zona di disponibilità): questo rappresenterà l'Outpost, utilizzando un ARN (Nome risorsa Amazon) e ilAWS numero risorsa.
- Nome Outpost: il nome dell' Outpost AWS.

Utilizzo degli Outposts con il CLI AWS

È possibile utilizzare AWS Command Line Interface (AWS CLI) per controllare più servizi AWS dalla riga di comando e automatizzarli tramite script. È possibile utilizzare la AWS (CLI) per operazioni ad hoc (una tantum).

Download e configurazione dell' AWS CLI

L'AWS CLI funziona in Windows, macOS o Linux. Utilizzare la procedura seguente per scaricarla e configurarla.

Per scaricare, installare e configurare la CLI

1. Scaricare la CLI AWS dalla pagina Web [AWSInterfaccia a riga di comando](#).
2. Seguire le istruzioni per [l'installazione della CLI AWS](#) e la [configurazione della AWS CLI](#) nella AWS Command Line Interfaceguida per l'utente.

Utilizzo di AWS CLI con Outposts

Utilizzare la seguente operazione CLI per creare un cluster di cache che utilizza Outposts:

- [create-cache-cluster](#)— Utilizzando questa operazione, il `outpost-mode` accetta un valore che specifica se i nodi nel cluster di cache vengono creati in un singolo Outpost o su più Outposts.

Note

Al momento, solo `single-outpost` Modalità supportata.

```
aws elasticache create-cache-cluster \  
  --cache-cluster-id cache cluster id \  
  --outpost-mode single-outpost \  
  \
```

Progettazione e gestione del proprio cluster ElastiCache per l'implementazione Memcached

Se hai bisogno di un controllo granulare sul cluster ElastiCache, puoi scegliere di progettare il tuo cluster. ElastiCache consente di usare un cluster basato su nodi, scegliendo il tipo di nodo, il numero di nodi e il posizionamento dei nodi nelle zone di disponibilità AWS del cluster. Poiché ElastiCache è un servizio completamente gestito, automaticamente fornisce il provisioning dell'hardware, il monitoraggio, la sostituzione dei nodi e l'applicazione delle patch al software del cluster.

Per informazioni sulla configurazione, consulta [Configurazione](#). Per i dettagli sulla gestione, l'aggiornamento o l'eliminazione di nodi o cluster, consulta [Gestione di nodi](#). Per una panoramica dei componenti principali di un'implementazione Amazon ElastiCache quando si progetta autonomamente un cluster ElastiCache, consulta questi [concetti chiave](#).

Argomenti

- [ElastiCache per componenti e funzionalità Memcached](#)
- [Gestione dei cluster](#)
- [Gestione di nodi](#)

ElastiCache per componenti e funzionalità Memcached

Di seguito, puoi trovare una panoramica dei componenti principali di una distribuzione Amazon ElastiCache for Memcached.

Argomenti

- [ElastiCache nodi](#)
- [ElastiCache per cluster Memcached](#)
- [AWS Regioni e zone di disponibilità](#)
- [ElastiCache per gli endpoint Memcached](#)
- [ElastiCache gruppi di parametri](#)
- [ElastiCache sicurezza](#)
- [ElastiCache gruppi di sottoreti](#)
- [ElastiCache per gli eventi Memcached](#)

ElastiCache nodi

Un nodo è l'elemento costitutivo più piccolo di una ElastiCache distribuzione. Un nodo può essere isolato o avere qualche tipo di rapporto con altri nodi.

Un nodo è un blocco di dimensioni fisse di RAM collegata alla rete. Ogni nodo esegue un'istanza di Memcached. Se necessario, puoi eseguire il dimensionamento verso l'alto o verso il basso dei nodi di un cluster in un tipo di istanza diverso. Per ulteriori informazioni, consulta [Scalabilità ElastiCache per Memcached](#).

Ogni nodo all'interno di un cluster è dello stesso tipo di istanza ed esegue lo stesso motore di cache. Ogni nodo di cache dispone del proprio Domain Name Service (DNS) e della propria porta. Sono supportati più tipi di nodi di cache, ciascuno con quantità variabili di memoria associata. Per una lista di tipi di istanze di nodo supportate, consulta [Tipi di nodi supportati](#).

Puoi acquistare nodi su pay-as-you-go base volontaria, pagando solo per l'utilizzo di un nodo. Oppure puoi acquistare nodi riservati a una tariffa oraria significativamente ridotta. Se il tasso di utilizzo è elevato, l'acquisto di nodi riservati aiuta a risparmiare. Supponiamo che il cluster sia quasi sempre in uso e che di tanto in tanto vengano aggiunti nodi per gestire i picchi di utilizzo. In questo caso, puoi acquistare diversi nodi riservati da utilizzare per la maggior parte del tempo e acquistare pay-as-you-

go nodi per le volte in cui occasionalmente devi aggiungere nodi. Per ulteriori informazioni sui nodi riservati, consulta [Nodi riservati ElastiCache](#).

Il motore Memcached supporta la funzione Individuazione automatica. La funzione Individuazione automatica consente ai programmi client di identificare automaticamente tutti i nodi di un cluster di cache e di avviare e mantenere le connessioni a tutti questi nodi. Con l'individuazione automatica non occorre connettere manualmente l'applicazione ai singoli nodi. L'applicazione si connette invece a un endpoint di configurazione. La voce DNS dell'endpoint di configurazione contiene le voci CNAME di ciascuno degli endpoint dei nodi di cache. Di conseguenza, connettendosi all'endpoint di configurazione, l'applicazione dispone immediatamente di informazioni su tutti i nodi del cluster e può connettersi a ognuno di loro. Non occorre codificare gli endpoint dei singoli nodi di cache nell'applicazione. Per ulteriori informazioni, consulta [Individuazione automatica](#).

Per ulteriori informazioni sui nodi, consulta [Gestione di nodi](#).

ElastiCache per cluster Memcached

Un cluster Memcached è un raggruppamento logico di uno o più [ElastiCache nodi](#). In un cluster Memcached i dati vengono partizionati tra i nodi.

Molte ElastiCache operazioni sono rivolte ai cluster:

- Creazione di un cluster
- Modifica di un cluster
- Eliminazione di un cluster
- Visualizzazione degli elementi in un cluster
- Aggiunta o rimozione di tag di allocazione costi a e da un cluster

Per informazioni più dettagliate, consulta i seguenti argomenti correlati:

- [Gestione dei cluster](#) e [Gestione di nodi](#)

Informazioni su cluster, nodi e operazioni correlate.

- [AWS limiti di servizio: Amazon ElastiCache](#)

Informazioni sui ElastiCache limiti, come il numero massimo di nodi o cluster.

Se devi superare questi limiti, effettua la richiesta utilizzando il [modulo di richiesta del nodo di ElastiCache cache Amazon](#).

- [Limitazione dell'impatto degli errori](#)

Informazioni su come migliorare la tolleranza ai guasti dei cluster.

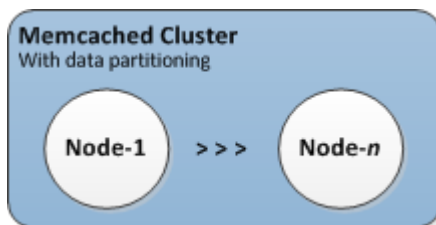
Configurazioni tipiche dei cluster

Memcached supporta fino a 300 nodi per cliente per ogni AWS regione, con ogni cluster con 1-60 nodi. Esegui la partizione dei dati tra i nodi di un cluster Memcached.

Quando si utilizza il motore Memcached, i cluster possono essere composti da 1—60 nodi. Esegui la partizione del database tra i nodi. L'applicazione esegue operazioni di lettura e di scrittura sull'endpoint di ciascun nodo. Per ulteriori informazioni, consulta [Individuazione automatica](#).

Per una migliore tolleranza agli errori, posiziona i nodi Memcached in varie zone di disponibilità (AZ) all'interno della regione del cluster. AWS In questo modo, un guasto in una zona di disponibilità ha un impatto minimo sul cluster e sull'applicazione. Per ulteriori informazioni, consulta [Limitazione dell'impatto degli errori](#).

Quando cambia la domanda per il cluster Memcached, puoi aumentare o diminuire le dimensioni del cluster aggiungendo o rimuovendo nodi che partizionano nuovamente i dati nella nuova quantità di nodi. Ti consigliamo di utilizzare un hashing coerente quando partizioni i dati. Per ulteriori informazioni sull'hashing coerente, consulta [Configurazione del client ElastiCache per un efficiente bilanciamento del carico](#). Nel diagramma seguente sono disponibili esempi di cluster Memcached a nodo singolo e a più nodi.



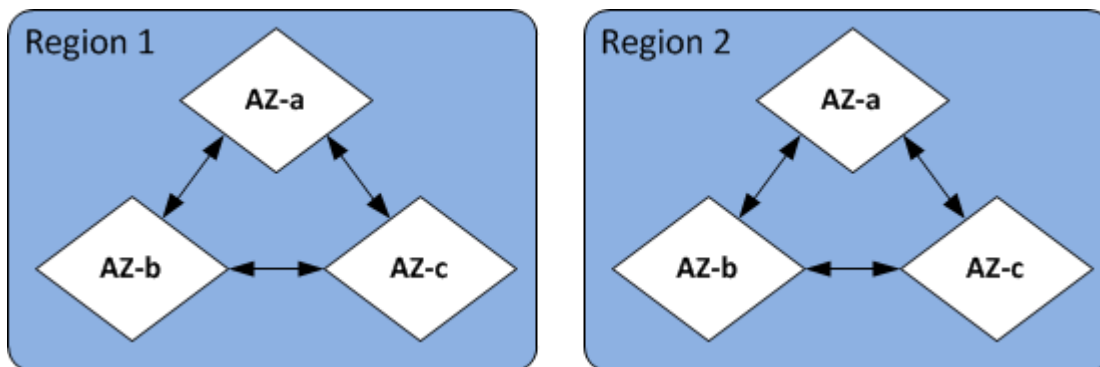
AWS Regioni e zone di disponibilità

Amazon ElastiCache for Memcached è disponibile in diverse AWS regioni del mondo. In questo modo, puoi avviare ElastiCache cluster nelle località che soddisfano i tuoi requisiti aziendali. Ad esempio, puoi lanciare nella AWS regione più vicina ai tuoi clienti o per soddisfare determinati requisiti legali.

Per impostazione predefinita, gli AWS SDK AWS CLI, l' ElastiCache API e la ElastiCache console fanno riferimento alla regione Stati Uniti occidentali (Oregon). Man mano ElastiCache che la

disponibilità viene estesa a nuove AWS regioni, sono disponibili anche nuovi endpoint per queste AWS regioni da utilizzare nelle richieste HTTP, negli AWS SDK e nella console. AWS CLI ElastiCache

Ogni AWS regione è progettata per essere completamente isolata dalle altre regioni. AWS All'interno di ciascuna regione sono presenti più zone di disponibilità. Avviando i nodi in diverse zone di disponibilità, puoi ottenere la massima tolleranza ai guasti possibile. Per ulteriori informazioni su AWS regioni e zone di disponibilità, vedere [Scelta di regioni e zone di disponibilità](#).



Per informazioni sulle AWS regioni supportate da ElastiCache e sui relativi endpoint, consulta [Regioni ed endpoint supportati](#).

ElastiCache per gli endpoint Memcached

Un endpoint è l'indirizzo univoco utilizzato dall'applicazione per connettersi a un ElastiCache nodo o cluster.

Ciascun nodo di un cluster Memcached ha il proprio endpoint. Anche il cluster ha un proprio endpoint, chiamato endpoint di configurazione. Se abiliti la funzione Individuazione automatica e ti connetti all'endpoint di configurazione, la tua applicazione conosce automaticamente l'endpoint di ciascun nodo, anche dopo l'aggiunta o la rimozione di nodi dal cluster. Per ulteriori informazioni, consulta [Individuazione automatica](#).

Per ulteriori informazioni, consulta la pagina relativa agli [endpoint](#).

ElastiCache gruppi di parametri

I gruppi di parametri cache sono un modo facile per gestire le impostazioni di runtime per il software di motore supportato. I parametri vengono utilizzati per controllare l'utilizzo della memoria, le policy di espulsione, le dimensioni degli elementi e altro ancora. Un gruppo di ElastiCache parametri è una

raccolta denominata di parametri specifici del motore che è possibile applicare a un cluster. In questo modo, sei certo che tutti i nodi di quel cluster siano configurati esattamente nello stesso modo.

Per un elenco dei parametri supportati, dei valori predefiniti e di quelli che si possono modificare, consulta [DescribeEngineDefaultParameters](#) ([describe-engine-default-parameters](#)).

Per informazioni più dettagliate sui gruppi di ElastiCache parametri, vedere. [Configurazione dei parametri di motore con i gruppi di parametri](#)

ElastiCache sicurezza

Per una maggiore sicurezza, l'accesso ai ElastiCache nodi è limitato alle applicazioni in esecuzione su istanze Amazon EC2 inserite nella whitelist. Puoi controllare le istanze Amazon EC2 che possono accedere al cluster utilizzando i gruppi di sicurezza.

Per impostazione predefinita, tutti i nuovi ElastiCache cluster vengono lanciati in un ambiente Amazon Virtual Private Cloud (Amazon VPC). Puoi utilizzare i gruppi di sottoreti per concedere l'accesso al cluster dalle istanze Amazon EC2 in esecuzione su sottoreti specifiche. Se scegliere di eseguire il cluster al di fuori del Amazon VPC, puoi creare Gruppi di sicurezza per autorizzare istanze Amazon EC2 in esecuzione all'interno di gruppi di sicurezza Amazon EC2 specifici.

ElastiCache gruppi di sottoreti

Un gruppo di sottoreti è una raccolta di sottoreti (generalmente private) che è possibile designare per i cluster in esecuzione in un ambiente cloud privato virtuale (Amazon VPC).

Se crei un cluster in un Amazon VPC, devi specificare un gruppo di sottoreti della cache. ElastiCache utilizza quel gruppo di sottorete di cache per scegliere una sottorete e gli indirizzi IP all'interno di quella sottorete da associare ai nodi della cache.

Per ulteriori informazioni sull'utilizzo del gruppo di sottoreti della cache in un ambiente Amazon VPC, consulta [Amazon VPC e sicurezza di ElastiCache](#), [Authorize access](#) e [Sottoreti e gruppi di sottoreti](#).

ElastiCache per gli eventi Memcached

Quando si verificano eventi significativi in un cluster di cache, ElastiCache invia una notifica a un argomento specifico di Amazon SNS. Gli eventi significativi possono includere, ad esempio, l'impossibilità di aggiungere un nodo, l'aggiunta di un nodo, la modifica di un gruppo di sicurezza e altro ancora. Monitorando gli eventi chiave, puoi conoscere lo stato corrente dei cluster e intraprendere operazioni correttive a seconda dell'evento.

Per ulteriori informazioni sugli ElastiCache eventi, consulta [Monitoraggio di Amazon SNS per gli eventi ElastiCache](#).

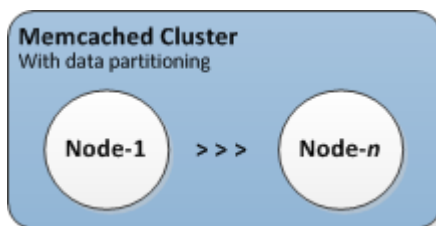
Gestione dei cluster

Un cluster è una raccolta di uno o più nodi di cache, ognuno dei quali esegue un'istanza del software del motore di cache Memcached. Quando si crea un cluster, occorre specificare il motore e la versione che verrà utilizzata da tutti i nodi.

Il diagramma seguente illustra un tipico cluster Memcached. I cluster Memcached contengono da 1 a 60 nodi su cui partizionare orizzontalmente i dati.

Per richiedere un aumento dei limiti, consulta [AWS Limiti del servizio](#) e selezionare il tipo di limite Nodi per cluster per tipo di istanza.

Un tipico cluster Memcached appare come segue.



La maggior parte delle ElastiCache operazioni viene eseguita a livello di cluster. Un cluster può essere configurato con un numero specifico di nodi e con un gruppo di parametri che controlla le proprietà di ciascun nodo. Tutti i nodi all'interno di un cluster sono progettati in modo da essere dello stesso tipo e da avere le stesse impostazioni del gruppo di parametri e del gruppo di sicurezza.

Ogni cluster deve avere un proprio identificatore. L'identificatore del cluster è un nome fornito dal cliente. Questo identificatore specifica un particolare cluster quando interagisce con l'ElastiCache API e i comandi. AWS CLI L'identificatore del cluster deve essere univoco per quel cliente in una regione. AWS

ElastiCache supporta più versioni del motore. A meno che non ci siano motivi specifici, consigliamo di utilizzare l'ultima versione.

ElastiCache i cluster sono progettati per essere accessibili utilizzando un'istanza Amazon EC2. Se il cluster viene avviato in un virtual private cloud (VPC) basato sul servizio Amazon VPC, vi si può accedere dall'esterno di AWS. Per ulteriori informazioni, consulta [Modelli di accesso a una cache ElastiCache in un Amazon VPC](#).

[Per un elenco delle versioni Memcached supportate, consulta Supportato per le versioni Memcached. ElastiCache](#)

Scelta del tipo di rete

ElastiCache supporta Internet Protocol versioni 4 e 6 (IPv4 e IPv6), consentendo di configurare il cluster per accettare:

- solo connessioni IPv4,
- solo connessioni IPv6,
- connessioni IPv4 e IPv6 (dual-stack)

IPv6 è supportato per i carichi di lavoro che utilizzano il motore Memcached dalla versione 1.6.6 in avanti su tutte le istanze sviluppate sul [Nitro system](#). Non sono previsti costi aggiuntivi per l'accesso a ElastiCache su IPv6.

Note

La migrazione dei cluster creati prima della disponibilità di IPV6/dual-stack non è supportata. Anche il passaggio da un tipo di rete all'altro nei nuovi cluster creati non è supportato.

Configurazione di sottoreti per tipo di rete

Se crei un cluster in un Amazon VPC, devi specificare un gruppo di sottoreti. ElastiCache utilizza il gruppo di sottoreti per selezionare una sottorete e gli indirizzi IP all'interno di quella sottorete da associare ai nodi. I cluster ElastiCache richiedono una sottorete dual-stack con indirizzi IPv4 e IPv6 assegnati agli stessi per funzionare in modalità dual-stack e una sottorete solo IPv6 per funzionare come solo IPv6.

Utilizzo di dual-stack

Quando si crea un cluster di cache e si sceglie dual-stack come tipo di rete, è necessario specificare un tipo di individuazione IP: IPv4 o IPv6. ElastiCache imposterà automaticamente il tipo di rete e l'individuazione IP su IPv6, ma questa impostazione può essere modificata. Se si utilizza Individuazione automatica, solo gli indirizzi IP del tipo IP scelto vengono restituiti al client Memcached.

Per mantenere la compatibilità con le versioni precedenti per tutti i client esistenti, viene introdotta l'individuazione IP, che consente di selezionare il tipo di IP (ad esempio IPv4 o IPv6) da pubblicizzare nel protocollo di individuazione. Sebbene ciò limiti l'individuazione automatica a un solo tipo di

IP, il dual-stack è comunque vantaggioso grazie a Individuazione automatica, in quanto consente migrazioni (o rollback) da un tipo di individuazione IP IPv4 a uno IPv6 senza tempi di inattività.

Cluster ElastiCache dual-stack abilitati per TLS

Quando TLS è abilitato per i cluster ElastiCache, le funzioni di individuazione del cluster `config get cluster` restituiscono nomi host anziché IP. I nomi host vengono quindi utilizzati al posto degli IP per connettersi al cluster ElastiCache ed eseguire un handshake TLS. Ciò significa che i client non saranno interessati dal parametro Individuazione IP. Per i cluster abilitati per TLS, il parametro Individuazione IP non ha alcun effetto sul protocollo IP preferito. Invece, il protocollo IP utilizzato verrà determinato in base a quello preferito dal client durante la risoluzione dei nomi host DNS.

Per esempi su come configurare una preferenza protocollo IP durante la risoluzione dei nomi host DNS, consulta [Cluster dual stack abilitati per TLS ElastiCache](#).

Utilizzo di AWS Management Console

Durante la creazione di un cluster di cache mediante la AWS Management Console, in Connectivity (Connettività), scegli un tipo di rete, IPv4, IPv6 o Dual stack. Se si sceglie dual-stack, è necessario selezionare un tipo di individuazione IP: IPv6 o IPv4.

Per ulteriori informazioni, consulta [Creazione di un cluster Memcached \(console\)](#).

Utilizzo della CLI

Durante la creazione di un cluster di cache mediante l'interfaccia a riga di comando, si utilizza il comando [create-cache-cluster](#) e si specificano i parametri `NetworkType` e `IPDiscovery`:

Per Linux, macOS o Unix:

```
aws elasticache create-cache-cluster \  
  --cache-cluster-id "cluster-test" \  
  --engine memcached \  
  --cache-node-type cache.m5.large \  
  --num-cache-nodes 1 \  
  --network-type dual_stack \  
  --ip-discovery ipv4
```

Per Windows:

```
aws elasticache create-cache-cluster ^
  --cache-cluster-id "cluster-test" ^
  --engine memcached ^
  --cache-node-type cache.m5.large ^
  --num-cache-nodes 1 ^
  --network-type dual_stack ^
  --ip-discovery ipv4
```

Identificazione automatica dei nodi nel cluster

Per cluster che eseguono il motore Memcached, ElastiCache supporta la funzione Individuazione automatica che consente ai programmi client di identificare automaticamente tutti i nodi di un cluster di cache e di avviare e mantenere le connessioni a tutti questi nodi.

Note

Individuazione automatica viene aggiunta per cluster di cache in esecuzione su Amazon ElastiCache Memcached.

Con Individuazione automatica, non occorre collegare manualmente l'applicazione ai singoli nodi di cache; invece, l'applicazione si collega a un nodo Memcached e recupera l'elenco di nodi. Da tale elenco l'applicazione è in grado di conoscere il resto dei nodi nel cluster e può connettersi a uno qualsiasi di essi. Non è necessario codificare gli endpoint dei singoli nodi di cache nell'applicazione.

Se si utilizza il tipo di rete dual-stack nel cluster, Individuazione automatica restituisce solo indirizzi IPv4 o IPv6, a seconda di quello selezionato. Per ulteriori informazioni, consulta [Scelta del tipo di rete](#).

Tutti i nodi di cache nel cluster mantengono un elenco di metadati relativi a tutti gli altri nodi. Questi metadati vengono aggiornati ogni volta che si aggiungono o rimuovono nodi dal cluster.

Argomenti

- [Vantaggi di Individuazione automatica](#)
- [Funzionamento di Individuazione automatica](#)
- [Uso di Individuazione automatica](#)
- [Connessione manuale ai nodi di cache](#)

- [Aggiunta di Individuazione automatica alla libreria del client](#)
- [Client ElastiCache con individuazione automatica](#)

Vantaggi di Individuazione automatica

Individuazione automatica offre i seguenti vantaggi:

- Quando si incrementa il numero di nodi in un cluster di cache, i nuovi nodi vengono registrati automaticamente con l'endpoint di configurazione e con tutti gli altri nodi. Quando si rimuovono nodi dal cluster di cache, i nodi che vengono a mancare annullano automaticamente la loro registrazione. In entrambi i casi, tutti gli altri nodi nel cluster vengono aggiornati con i metadati di nodi di cache più recenti.
- Guasti nei nodi di cache vengono rilevati automaticamente; nodi non riusciti vengono sostituiti automaticamente.

Note

Finché la sostituzione del nodo non viene completata, il nodo continuerà a restituire un errore.

- Un programma client deve solo connettersi all'endpoint di configurazione. Successivamente, la libreria di Individuazione automatica si collega a tutti gli altri nodi nel cluster.
- I programmi client eseguono il polling del cluster una volta il minuto (questo intervallo può essere modificato se necessario). In caso di modifiche alla configurazione del cluster, ad esempio nodi nuovi o eliminati, il client riceve un elenco aggiornato di metadati. Quindi il client si collega a, o si scollega da, questi nodi in base alle esigenze.

Individuazione automatica è abilitata su tutti i cluster di cache ElastiCache Memcached. Per utilizzare questa caratteristica non è necessario riavviare i nodi di cache.

Funzionamento di Individuazione automatica

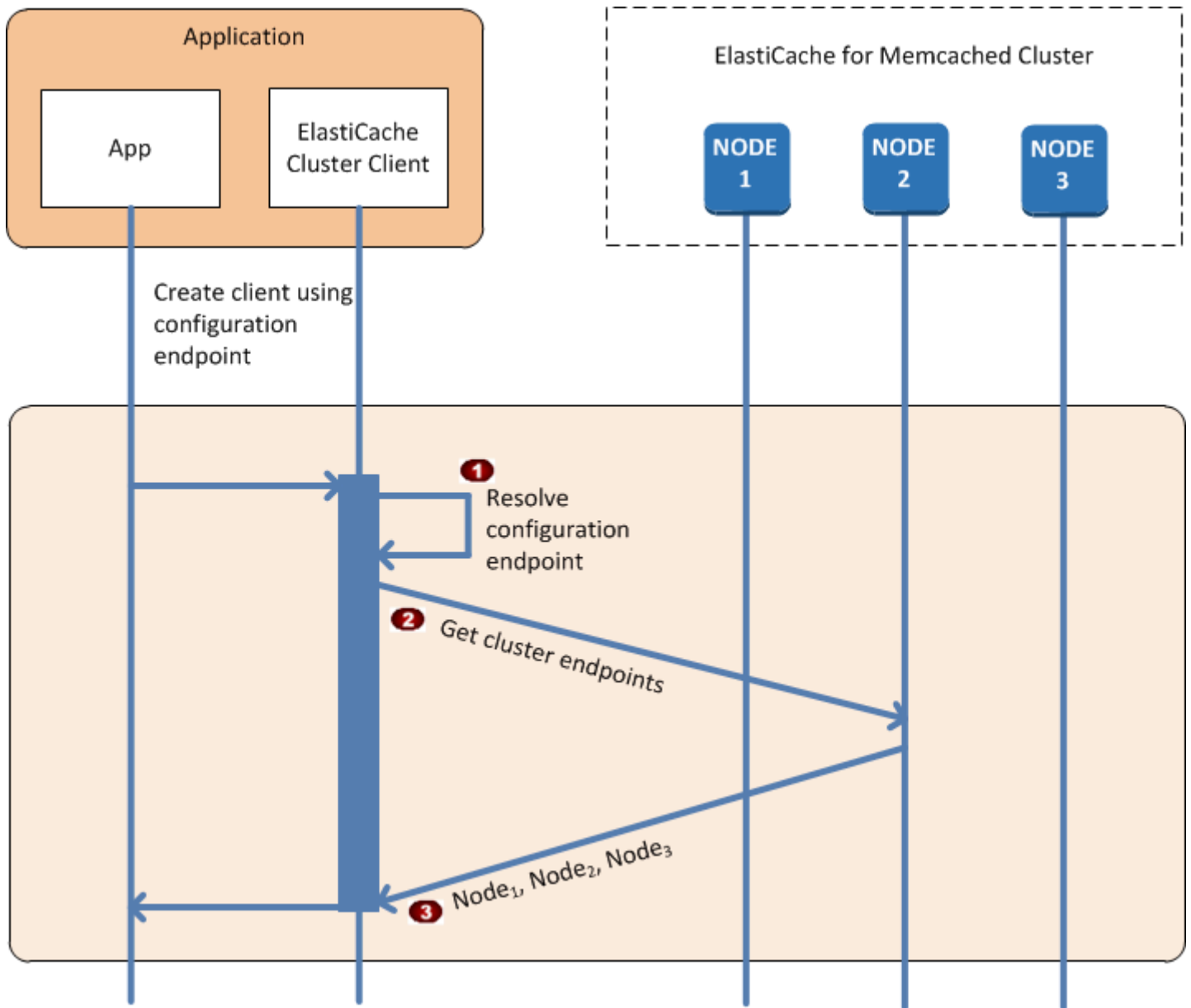
Argomenti

- [Connessione ai nodi di cache](#)
- [Operazioni cluster normali](#)
- [Altre operazioni](#)

In questa sezione viene descritto in che modo le applicazioni client utilizzano Client del cluster ElastiCache per gestire le connessioni ai nodi di cache e interagire con gli item di dati nella cache.

Connessione ai nodi di cache

Dal punto di vista dell'applicazione, la connessione all'endpoint di configurazione del cluster non è diverso dalla connessione diretta a un singolo nodo di cache. Il diagramma di sequenza seguente mostra il processo di connessione ai nodi di cache.



Processo di connessione ai nodi di cache

- L'applicazione risolve il nome DNS dell'endpoint di configurazione. Poiché l'endpoint di configurazione mantiene le voci CNAME per tutti i nodi di cache, il nome DNS viene risolto in uno dei nodi; il client può quindi connettersi a tale nodo.
- Il client richiede le informazioni di configurazione per tutti gli altri nodi. Poiché ogni nodo mantiene informazioni di configurazione per tutti i nodi nel cluster, qualsiasi nodo può passare informazioni di configurazione al client su richiesta.

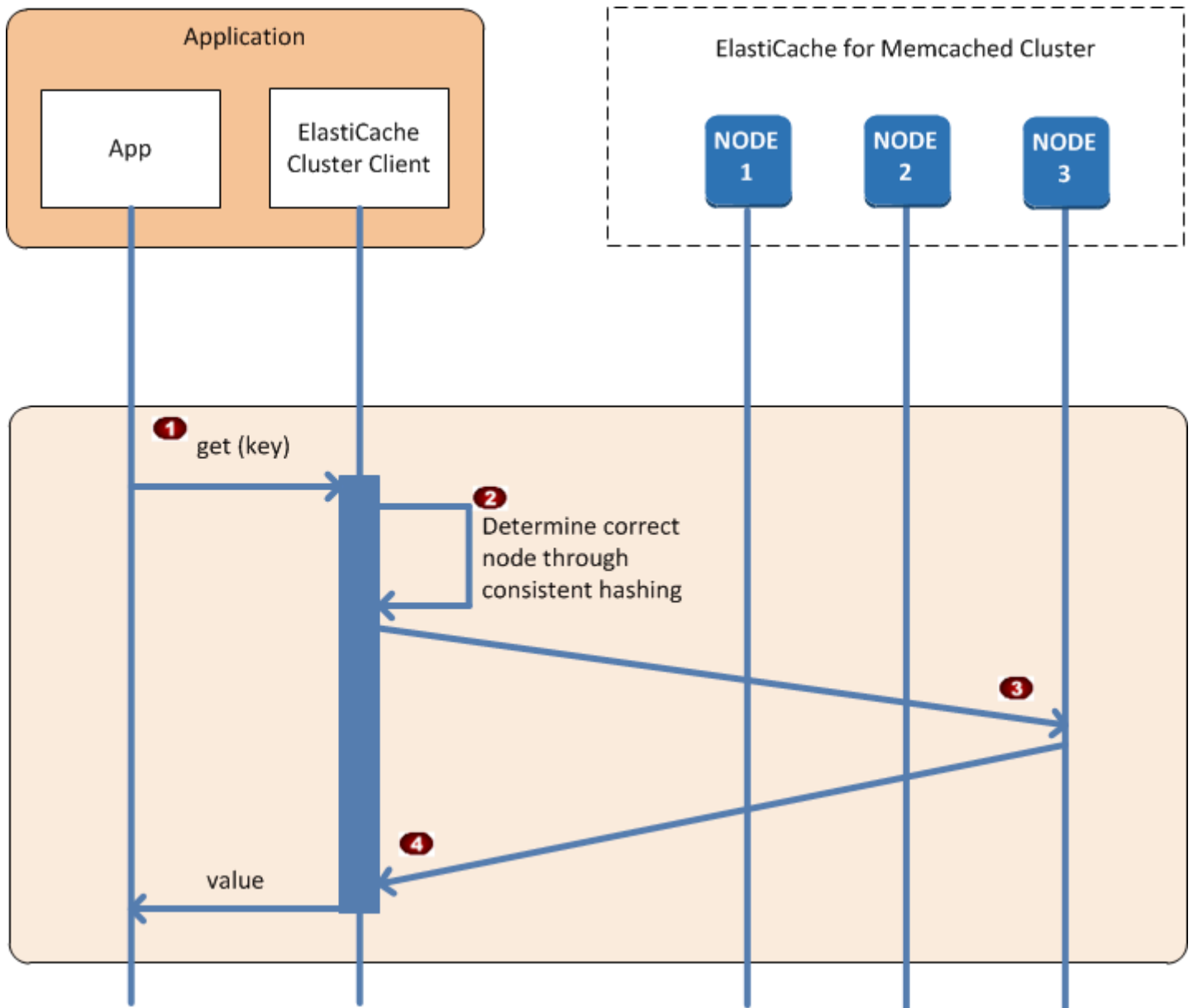
- Il client riceve l'elenco corrente di nomi host e indirizzi IP dei nodi di cache. Può quindi connettersi a tutti gli altri nodi nel cluster.

Note

Il programma client aggiorna il suo elenco di nomi host e indirizzi IP dei nodi di cache una volta il minuto. Questo intervallo di polling può essere modificato se necessario.

Operazioni cluster normali

Quando l'applicazione si è connessa a tutti i nodi di cache, il Client cluster ElastiCache determina quali nodi devono archiviare singoli item di dati e su quali nodi devono essere eseguite query per tali item di dati in seguito. Il diagramma di sequenza seguente mostra il processo di operazioni cluster normali.



Processo di operazioni cluster normali

- L'applicazione emette una richiesta get per un particolare item di dati, identificato dalla sua chiave.
- Il client utilizza un algoritmo di hashing rispetto alla chiave per determinare quale nodo di cache contiene l'item di dati.
- L'item di dati viene richiesto dal nodo appropriato.
- L'item di dati viene restituito all'applicazione.

Altre operazioni

In alcune situazioni, è possibile effettuare una modifica ai nodi di un cluster. Ad esempio, è possibile aggiungere un nodo aggiuntivo per soddisfare una domanda aggiuntiva oppure eliminare un nodo per risparmiare denaro durante i periodi di domanda ridotta. Oppure potresti sostituire un nodo a causa di un errore di un nodo di un tipo o di un altro.

In presenza di una modifica nel cluster che richiede un aggiornamento dei metadati negli endpoint del cluster, tale modifica viene apportata a tutti i nodi contemporaneamente. Pertanto, i metadati in qualsiasi nodo specificato sono coerenti con i metadati in tutti gli altri nodi nel cluster.

In ciascuno di questi casi, i metadati sono coerenti tra tutti i nodi in qualsiasi momento poiché i metadati vengono aggiornati contemporaneamente per tutti i nodi nel cluster. Si consiglia di utilizzare sempre l'endpoint di configurazione per ottenere gli endpoint dei diversi nodi nel cluster. L'uso dell'endpoint di configurazione evita che vengano ricevuti dati dell'endpoint da un nodo che "scompare".

Aggiunta di un nodo

Durante il periodo di attivazione del nodo, il suo endpoint non è incluso nei metadati. Non appena il nodo diventa disponibile, viene aggiunto ai metadati di ciascuno dei nodi del cluster. In questo scenario, i metadati sono coerenti tra tutti i nodi e sarai in grado di interagire con il nuovo nodo solo dopo che questo diventa disponibile. Il nodo non è noto prima che diventi disponibile e l'interazione con i nodi nel cluster avviene come se il nuovo nodo non esistesse.

Eliminazione di un nodo

Quando un nodo viene rimosso, il relativo endpoint viene rimosso innanzitutto dai metadati e quindi il nodo viene rimosso dal cluster. In questo scenario, i metadati in tutti i nodi sono coerenti e in nessun momento conterranno l'endpoint per il nodo che deve essere rimosso mentre il nodo non è disponibile. Durante la rimozione del nodo il tempo non è segnalato nei metadati e pertanto l'applicazione interagisce solo con gli $n-1$ nodi rimanenti, come se il nodo non esistesse.

Sostituzione di un nodo

Se un nodo restituisce un errore, ElastiCache arresta tale nodo e rende operativa una sostituzione. Il processo di sostituzione richiede alcuni minuti. Durante questo periodo di tempo i metadati in tutti i nodi mostrano ancora l'endpoint per il nodo non riuscito, ma qualsiasi tentativo di interagire con il nodo restituisce un errore. Pertanto, la logica deve sempre includere la logica di ripetizione.

Uso di Individuazione automatica

Per iniziare a utilizzare Individuazione automatica, segui le fasi sottostanti:

- [Fase 1: ottenere l'endpoint di configurazione](#)
- [Fase 2: scaricare il cluster del client ElastiCache](#)
- [Fase 3: modificare il programma dell'applicazione](#)

Fase 1: ottenere l'endpoint di configurazione

Per connettersi a un cluster, i programmi client devono conoscere l'endpoint di configurazione del cluster. Consulta l'argomento [Individuazione degli endpoint di un cluster \(console\)](#)

Puoi anche utilizzare il comando `aws elasticache describe-cache-clusters` con il parametro `--show-cache-node-info`:

Qualunque sia il metodo utilizzato per trovare gli endpoint del cluster, l'indirizzo dell'endpoint di configurazione conterrà sempre `.cfg`.

Example Individuazione di endpoint utilizzando la AWS CLI per ElastiCache

Per Linux, macOS o Unix:

```
aws elasticache describe-cache-clusters \  
  --cache-cluster-id mycluster \  
  --show-cache-node-info
```

Per Windows:

```
aws elasticache describe-cache-clusters ^  
  --cache-cluster-id mycluster ^  
  --show-cache-node-info
```

Questa operazione genera un output simile al seguente (formato JSON):

```
{  
  "CacheClusters": [  
    {  
      "Engine": "memcached",  
      "CacheNodes": [  
        {
```



```
    "CacheNodeId": "0001",
    "Endpoint": {
      "Port": 11211,
      "Address": "mycluster.fnjyzo.cfg.0001.use1.cache.amazonaws.com"
    },
    "CacheNodeStatus": "available",
    "ParameterGroupStatus": "in-sync",
    "CacheNodeCreateTime": "2016-10-12T21:39:28.001Z",
    "CustomerAvailabilityZone": "us-east-1e"
  },
  {
    "CacheNodeId": "0002",
    "Endpoint": {
      "Port": 11211,
      "Address": "mycluster.fnjyzo.cfg.0002.use1.cache.amazonaws.com"
    },
    "CacheNodeStatus": "available",
    "ParameterGroupStatus": "in-sync",
    "CacheNodeCreateTime": "2016-10-12T21:39:28.001Z",
    "CustomerAvailabilityZone": "us-east-1a"
  }
],
"CacheParameterGroup": {
  "CacheNodeIdsToReboot": [],
  "CacheParameterGroupName": "default.memcached1.4",
  "ParameterApplyStatus": "in-sync"
},
"CacheClusterId": "mycluster",
"PreferredAvailabilityZone": "Multiple",
"ConfigurationEndpoint": {
  "Port": 11211,
  "Address": "mycluster.fnjyzo.cfg.use1.cache.amazonaws.com"
},
"CacheSecurityGroups": [],
"CacheClusterCreateTime": "2016-10-12T21:39:28.001Z",
"AutoMinorVersionUpgrade": true,
"CacheClusterStatus": "available",
"NumCacheNodes": 2,
"ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/home#client-download:",
"CacheSubnetGroupName": "default",
"EngineVersion": "1.4.24",
"PendingModifiedValues": {},
"PreferredMaintenanceWindow": "sat:06:00-sat:07:00",
```

```
        "CacheNodeType": "cache.r3.large"
    }
]
}
```

Fase 2: scaricare il cluster del client ElastiCache

Per sfruttare Individuazione automatica, i programmi client devono utilizzare il Client del cluster ElastiCache. Il Client del cluster ElastiCache è disponibile per Java, PHP e .NET e contiene tutta la logica necessaria per l'individuazione e la connessione a tutti i nodi di cache.

Download del client del cluster ElastiCache

1. Accedi alla console di gestione AWS e apri la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Dalla console ElastiCache, scegli Client del cluster ElastiCache, quindi scegli Scarica.

Il codice fonte per il Client del cluster ElastiCache per Java è disponibile all'indirizzo <https://github.com/amazonwebservices/aws-elasticache-cluster-client-memcached-for-java>. Questa libreria è basata sul client Spymemcached di uso comune. Il Client del cluster ElastiCache viene rilasciato con licenza del software Amazon <https://aws.amazon.com/asl>. Sei libero di modificare il codice fonte come meglio credi. Puoi persino incorporare il codice in altre librerie Memcached open source o nel tuo codice client.

Note

Per utilizzare il Client del cluster ElastiCache per PHP, devi innanzitutto installarlo sull'istanza Amazon EC2. Per ulteriori informazioni, consulta [Installazione del client del cluster ElastiCache per PHP](#).

Per un client con supporto TLS, scarica il file binario con PHP versione 7.4 o superiore.

Per utilizzare il Client del cluster ElastiCache per .NET, devi innanzitutto installarlo sull'istanza Amazon EC2. Per ulteriori informazioni, consulta [Installazione del client del cluster ElastiCache per .NET](#).

Fase 3: modificare il programma dell'applicazione

Modifica il programma dell'applicazione in modo che utilizzi Individuazione automatica. Nelle seguenti sezioni viene illustrato come utilizzare il Client del cluster ElastiCache per Java, PHP e .NET.

⚠ Important

Durante la specifica dell'endpoint di configurazione del cluster, assicurati che l'indirizzo dell'endpoint contenga ".cfg" come mostrato qui. Non utilizzare un CNAME o un endpoint che non contiene ".cfg" al suo interno.

```
"mycluster.fnjyzo.cfg.use1.cache.amazonaws.com";
```

La mancata specifica esplicita dell'endpoint di configurazione del cluster determina la configurazione di un nodo specifico.

Uso del Client del cluster ElastiCache per Java

Il programma sottostante dimostra in che modo utilizzare il Client del cluster ElastiCache per connettersi a un endpoint di configurazione del cluster e aggiungere un elemento di dati alla cache. Utilizzando Individuazione automatica, il programma si connette a tutti i nodi nel cluster senza ulteriori interventi.

```
package com.amazon.elasticache;

import java.io.IOException;
import java.net.InetSocketAddress;

// Import the &AWS;-provided library with Auto Discovery support
import net.spy.memcached.MemcachedClient;

public class AutoDiscoveryDemo {

    public static void main(String[] args) throws IOException {

        String configEndpoint = "mycluster.fnjyzo.cfg.use1.cache.amazonaws.com";
        Integer clusterPort = 11211;

        MemcachedClient client = new MemcachedClient(
            new InetSocketAddress(configEndpoint,
                clusterPort));

        // The client will connect to the other cache nodes automatically.

        // Store a data item for an hour.
        // The client will decide which cache host will store this item.
```

```
        client.set("theKey", 3600, "This is the data value");
    }
}
```

Uso del Client del cluster ElastiCache per PHP

Il programma sottostante dimostra in che modo utilizzare il Client del cluster ElastiCache per connettersi a un endpoint di configurazione del cluster e aggiungere un elemento di dati alla cache. Utilizzando Individuazione automatica, il programma si connette a tutti i nodi in un cluster senza ulteriori interventi.

Per utilizzare il Client del cluster ElastiCache per PHP, devi innanzitutto installarlo sull'istanza Amazon EC2. Per ulteriori informazioni, consulta [Installazione del client del cluster ElastiCache per PHP](#)

```
<?php

/**
 * Sample PHP code to show how to integrate with the Amazon ElastiCache
 * Auto Discovery feature.
 */

/* Configuration endpoint to use to initialize memcached client.
 * This is only an example. */
$server_endpoint = "mycluster.fnjyzo.cfg.use1.cache.amazonaws.com";

/* Port for connecting to the ElastiCache cluster.
 * This is only an example */
$server_port = 11211;

/**
 * The following will initialize a Memcached client to utilize the Auto Discovery
 * feature.
 *
 * By configuring the client with the Dynamic client mode with single endpoint, the
 * client will periodically use the configuration endpoint to retrieve the current
 * cache
 * cluster configuration. This allows scaling the cache cluster up or down in number
 * of nodes
 * without requiring any changes to the PHP application.
 *
 * By default the Memcached instances are destroyed at the end of the request.
```

```
* To create an instance that persists between requests,
*   use persistent_id to specify a unique ID for the instance.
* All instances created with the same persistent_id will share the same connection.
* See http://php.net/manual/en/memcached.construct.php for more information.
*/
$dynamic_client = new Memcached('persistent-id');
$dynamic_client->setOption(Memcached::OPT_CLIENT_MODE,
Memcached::DYNAMIC_CLIENT_MODE);
$dynamic_client->addServer($server_endpoint, $server_port);

/**
 * Store the data for 60 seconds in the cluster.
 * The client will decide which cache host will store this item.
 */
$dynamic_client->set('key', 'value', 60);

/**
 * Configuring the client with Static client mode disables the usage of Auto Discovery
 * and the client operates as it did before the introduction of Auto Discovery.
 * The user can then add a list of server endpoints.
 */
$static_client = new Memcached('persistent-id');
$static_client->setOption(Memcached::OPT_CLIENT_MODE, Memcached::STATIC_CLIENT_MODE);
$static_client->addServer($server_endpoint, $server_port);

/**
 * Store the data without expiration.
 * The client will decide which cache host will store this item.
 */
$static_client->set('key', 'value');
?>
```

Per un esempio di utilizzo del client del cluster ElastiCache con TLS abilitato, consulta [Using in transit encryption with PHP and Memcached](#).

Utilizzare il client del cluster ElastiCache per .NET.

Note

Il client del cluster .NET ElastiCache è diventato obsoleto a partire da maggio 2022.

Il client .NET per ElastiCache è open source all'indirizzo <https://github.com/awslabs/elasticache-cluster-config-net>.

Le applicazioni .NET in genere recuperano le loro configurazioni dal file config. Di seguito viene riportato un file config dell'applicazione di esempio.

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <configSections>
    <section
      name="clusterclient"
      type="Amazon.ElastiCacheCluster.ClusterConfigSettings,
Amazon.ElastiCacheCluster" />
  </configSections>

  <clusterclient>
    <!-- the hostname and port values are from step 1 above -->
    <endpoint hostname="mycluster.fnjyzo.cfg.use1.cache.amazonaws.com"
port="11211" />
  </clusterclient>
</configuration>
```

Il programma C# sottostante dimostra come utilizzare il Client del cluster ElastiCache per connettersi a un endpoint di configurazione del cluster e aggiungere un item di dati alla cache. Utilizzando Individuazione automatica, il programma si connette a tutti i nodi in un cluster senza ulteriori interventi.

```
// *****
// Sample C# code to show how to integrate with the Amazon ElastiCache Auto Discovery
// feature.

using System;

using Amazon.ElastiCacheCluster;

using Enyim.Caching;
using Enyim.Caching.Memcached;

public class DotNetAutoDiscoveryDemo {

    public static void Main(String[] args) {
```

```
// instantiate a new client.
ElastiCacheClusterConfig config = new ElastiCacheClusterConfig();
MemcachedClient memClient = new MemcachedClient(config);

// Store the data for 3600 seconds (1hour) in the cluster.
// The client will decide which cache host will store this item.
memClient.Store(StoreMode.Set, 3600, "This is the data value.");

} // end Main

} // end class DotNetAutoDiscoverDemo
```

Connessione manuale ai nodi di cache

Se il programma client non utilizza Individuazione automatica, può connettersi manualmente a ciascuno dei nodi di cache. Questo è il comportamento di default per i client Memcached.

Puoi ottenere un elenco di nomi host e numeri porta dei nodi di cache dalla [AWSConsole di gestione](#). Puoi anche utilizzare il comando AWS CLI `aws elasticache describe-cache-clusters` con il parametro `--show-cache-node-info`.

Example

Il seguente frammento di codice Java mostra come connettersi a tutti i nodi in un cluster di cache di quattro nodi:

```
...  
  
ArrayList<String> cacheNodes = new ArrayList<String>(  
    Arrays.asList(  
        "mycachecluster.fnjyzo.0001.use1.cache.amazonaws.com:11211",  
        "mycachecluster.fnjyzo.0002.use1.cache.amazonaws.com:11211",  
        "mycachecluster.fnjyzo.0003.use1.cache.amazonaws.com:11211",  
        "mycachecluster.fnjyzo.0004.use1.cache.amazonaws.com:11211"));  
  
MemcachedClient cache = new MemcachedClient(AddrUtil.getAddresses(cacheNodes));  
  
...
```

Important

Se aumenti o riduci il cluster di cache aggiungendo o rimuovendo nodi, sarà necessario aggiornare l'elenco di nodi nel codice client.

Aggiunta di Individuazione automatica alla libreria del client

Le informazioni di configurazione per Individuazione automatica vengono archiviate in modo ridondante in ogni nodo del cluster di cache. Le applicazioni client possono eseguire la query di qualsiasi nodo di cache e ottenere le informazioni di configurazione per tutti i nodi nel cluster.

Il modo utilizzato dall'applicazione dipende dalla versione del motore della cache:

- Se la versione del motore della cache è 1.4.14 o successive, utilizza il comando `config`.
- Se la versione del motore della cache è precedente alla 1.4.14, utilizza il comando `get AmazonElastiCache:cluster`.

I risultati di questi due comandi sono identici e vengono descritti nella sezione [Formato di output](#) sottostante.

Versione del motore della cache 1.4.14 o successiva

Per versioni del motore della cache 1.4.14 o successive, utilizza il comando `config`. Questo comando è stato aggiunto ai protocolli ASCII e binario Memcached da ElastiCache e viene implementato nel Client del cluster ElastiCache . Se desideri utilizzare Individuazione automatica con un'altra libreria client, allora sarà necessario espandere tale libreria per supportare il comando `config`.

Note

La documentazione seguente è relativa al protocollo ASCII; tuttavia, il comando `config` supporta ASCII e binario. Se desideri aggiungere il support per Identificazione automatica utilizzando il protocollo binario, fai riferimento al [codice fonte per il Client del cluster ElastiCache](#).

Sintassi

```
config [sub-command] [key]
```

Opzioni

Nome	Descrizione	Obbligatorio
sub-command	Il sottocomando utilizzando per interagire con un nodo di cache. Per Individuazione automatica, questo sottocomando è <code>get</code> .	Sì
key	La chiave sotto cui viene archiviata la configurazione cluster. Per Individuazione automatica, questa chiave è denominata <code>cluster</code> .	Sì

Per ottenere le informazioni di configurazione del cluster, utilizza il seguente comando:

```
config get cluster
```

Versione del motore della cache precedente a 1.4.14

Per ottenere le informazioni di configurazione del cluster, utilizza il seguente comando:

```
get AmazonElastiCache:cluster
```

Note

Non manomettere la chiave "AmazonElastiCache:cluster", poiché essa contiene le informazioni di configurazione del cluster. Se questa chiave viene sovrascritta, è possibile che il client sia erroneamente configurato per un breve periodo di tempo (non più di 15 secondi) prima che ElastiCache aggiorni automaticamente e correttamente le informazioni di configurazione.

Formato di output

A prescindere che utilizzi `config get cluster` o `get AmazonElastiCache:cluster`, la risposta è costituita sempre da due righe:

- Il numero di versione delle informazioni di configurazione. Ogni volta che si aggiunge o rimuove un nodo dal cluster di cache, il numero di versione viene incrementato di uno.

- Un elenco di nodi di cache. Ogni nodo nell'elenco è rappresentato da un gruppo `hostname|ip-address|port` e ogni nodo è delimitato da uno spazio.

Alla fine di ciascuna riga vengono visualizzati un carattere di ritorno a capo e un carattere di avanzamento riga (CR + LF). La riga di dati contiene un carattere di avanzamento riga (LF) alla fine, cui viene aggiunto CR + LF. La riga di versione config è terminata da LF senza CR.

Un cluster di cache contenente tre nodi viene rappresentato come riportato di seguito:

```
configversion\n
hostname|ip-address|port hostname|ip-address|port hostname|ip-address|port\n\r\n
```

Ogni nodo viene mostrato con il CNAME e l'indirizzo IP privato. Il CNAME è sempre presente; se l'indirizzo IP privato non è disponibile, non verrà visualizzato; tuttavia, i caratteri pipe "|" verranno ancora stampati.

Example

Di seguito è riportato un esempio del payload restituito quando si esegue la query delle informazioni di configurazione:

```
CONFIG cluster 0 136\r\n
12\n
myCluster.pc4ldq.0001.use1.cache.amazonaws.com|10.82.235.120|11211
  myCluster.pc4ldq.0002.use1.cache.amazonaws.com|10.80.249.27|11211\n\r\n
END\r\n
```

Note

- La seconda riga indica che le informazioni di configurazione sono state modificate dodici volte finora.
- Nella terza riga, l'elenco dei nodi è in ordine alfabetico per nome host. Questo ordinamento deve essere in una sequenza diversa da quella attualmente in uso nell'applicazione client.

Client ElastiCache con individuazione automatica

In questa sezione viene illustrata l'installazione e la configurazione dei client PHP ElastiCache e .NET .

Argomenti

- [Installazione e compilazione di client del cluster](#)
- [Configurazione client ElastiCache](#)

Installazione e compilazione di client del cluster

In questa sezione viene descritta l'installazione, la configurazione e la compilazione dei client del cluster di individuazione automatica Amazon ElastiCache PHP e .NET.

Argomenti

- [Installazione del client del cluster ElastiCache per .NET](#)
- [Installazione del client del cluster ElastiCache per PHP](#)
- [Compilazione del codice fonte per il client del cluster ElastiCache per PHP](#)

Installazione del client del cluster ElastiCache per .NET

Note

Il client del cluster .NET ElastiCache è diventato obsoleto a partire da maggio 2022.

Il codice del client del cluster ElastiCache .NET è disponibile come open source all'indirizzo <https://github.com/awslabs/elasticache-cluster-config-net>.

In questa sezione viene descritto come installare, aggiornare e rimuovere i componenti .NET per il client del cluster ElastiCache su istanze Amazon EC2. Per ulteriori informazioni sull'individuazione automatica, consulta [Auto discovery](#). Per un esempio di codice .NET per utilizzare il client, consulta [Auto discovery with DotNET](#).

Argomenti

- [Installazione di .NET](#)
- [Download del client del cluster ElastiCache .NET per ElastiCache](#)
- [Installazione di assembly AWS con NuGet](#)

Installazione di .NET

Per utilizzare l'SDK AWS .NET per ElastiCache, deve essere installato .NET 3.5 o versioni successive. In caso contrario, puoi scaricare e installare la versione più recente da <http://www.microsoft.com/net>.

Download del client del cluster ElastiCache .NET per ElastiCache

Download del client del cluster ElastiCache .NET per ElastiCache

1. Accedi alla AWS Management Console e apri la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nel riquadro di navigazione, fai clic su Client del cluster ElastiCache.
3. Nell'elenco Scarica client del cluster ElastiCache Memcached, scegli .NET, quindi fai clic su Scarica.

Installazione di assembly AWS con NuGet

NuGet è un sistema di gestione dei pacchetti per la piattaforma .NET. NuGet rileva le dipendenze dell'assembly e installa automaticamente tutti i file necessari. Gli assembly NuGet installati vengono archiviati con la soluzione, anziché in una posizione centrale come Program Files, pertanto puoi installare versioni specifiche per un'applicazione senza creare problemi di compatibilità.

Installazione di NuGet

NuGet può essere installato da Installation Gallery su MSDN; consulta <https://visualstudiogallery.msdn.microsoft.com/27077b70-9dad-4c64-adcf-c7cf6bc9970c>. Se stai utilizzando Visual Studio 2010 o versioni successive, NuGet viene installato automaticamente.

Puoi utilizzare NuGet da Esplora soluzioni o Console di Gestione pacchetti.

Utilizzo di NuGet da Esplora soluzioni

Per utilizzare NuGet da Esplora soluzioni in Visual Studio 2010

1. Dal menu Strumenti, scegli Gestione pacchetti libreria.
2. Fai clic su Console di Gestione pacchetti.

Per utilizzare NuGet da Esplora soluzioni in Visual Studio 2012 o Visual Studio 2013

1. Dal menu Strumenti, scegli Gestione pacchetti NuGet.
2. Fai clic su Console di Gestione pacchetti.

Dalla riga di comando, è possibile installare gli assembly utilizzando `Install-Package`, come mostrato di seguito.

```
Install-Package Amazon.ElastiCacheCluster
```

Per visualizzare una pagina per ogni pacchetto disponibile tramite NuGet, ad esempio gli assembly SDKAWS e .Extensions assemblies AWS, visita il sito Web NuGet all'indirizzo <http://www.nuget.org>. La pagina per ogni pacchetto include una riga di comando di esempio per l'installazione del pacchetto utilizzando la console e un elenco delle versioni precedenti del pacchetto disponibili tramite NuGet.

Per ulteriori informazioni sui comandi di Package Manager Console (Console di Gestione pacchetti), consulta <http://nuget.codeplex.com/wikipage?title=Package%20Manager%20Console%20Command%20Reference%20%28v1.3%29>.

Installazione del client del cluster ElastiCache per PHP

In questa sezione viene descritto come installare, aggiornare e rimuovere i componenti PHP per il client del cluster ElastiCache su istanze Amazon EC2. Per ulteriori informazioni sull'individuazione automatica, consulta [Identificazione automatica dei nodi nel cluster](#). Per il codice PHP di esempio per utilizzare il client, consulta [Uso del Client del cluster ElastiCache per PHP](#).

Argomenti

- [Download del pacchetto di installazione](#)
- [Per utenti con già installata l'estensione php-memcached](#)
- [Fasi di installazione per nuovi utenti](#)
- [Rimozione del client del cluster PHP](#)

Download del pacchetto di installazione

Per garantire l'uso della versione corretta del client del cluster ElastiCache per PHP, occorre conoscere la versione PHP installata sull'istanza Amazon EC2. Inoltre, è necessario sapere se l'istanza Amazon EC2 esegue una versione a 64 bit o a 32 bit di Linux.

Per determinare la versione PHP installata sull'istanza Amazon EC2

- Dal prompt dei comandi, eseguire il seguente comando:

```
php -v
```

La versione PHP verrà visualizzata nell'output, come nel seguente esempio:

```
PHP 5.4.10 (cli) (built: Jan 11 2013 14:48:57)
Copyright (c) 1997-2012 The PHP Group
Zend Engine v2.4.0, Copyright (c) 1998-2012 Zend Technologies
```

Note

Se le versioni PHP e Memcached non sono compatibili, verrà restituito un messaggio di errore simile al seguente:

```
PHP Warning: PHP Startup: memcached: Unable to initialize module
Module compiled with module API=20100525
```

```
PHP compiled with module API=20131226
These options need to match
in Unknown on line 0
```

In tal caso, occorre compilare il modulo dal codice fonte. Per ulteriori informazioni, consulta [Compilazione del codice fonte per il client del cluster ElastiCache per PHP](#).

Per determinare l'architettura dell'AMI Amazon EC2 (64 bit o 32 bit)

1. Accedi a AWS Management Console e apri la console Amazon EC2 all'indirizzo <https://console.aws.amazon.com/ec2/>.
2. Nell'elenco Istanze, fare clic sull'istanza Amazon EC2.
3. Nella scheda Description (Descrizione), cercare il campo AMI:. Un'istanza a 64 bit deve includere x86_64 come parte della descrizione; per un'istanza a 32 bit, cercare i386 o i686 in questo campo.

Puoi ora scaricare il client del cluster ElastiCache.

Download del client del cluster ElastiCache per PHP

1. Accedi alla AWS Management Console e apri la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Dalla console ElastiCache, scegli client cluster ElastiCache.
3. Dall'elenco Scarica client del cluster ElastiCache Memcached, scegli il client del cluster che corrisponde alla versione PHP e all'architettura dell'AMI, quindi scegli il pulsante Scarica.

Per un client con supporto TLS, scarica il file binario con PHP versione 7.4 o superiore.

Per utenti con già installata l'estensione php-memcached

Per aggiornare l'installazione **php-memcached**

1. Rimuovere l'installazione precedente dell'estensione Memcached per PHP come descritto dall'argomento [Rimozione del client del cluster PHP](#).
2. Installare la nuova estensione ElastiCache di php-memcached come descritto in precedenza in [Fasi di installazione per nuovi utenti](#).

Fasi di installazione per nuovi utenti

Argomenti

- [Installazione di PHP 7.x - 8.x per nuovi utenti](#)
- [Installazione di PHP 5.x per nuovi utenti](#)

Installazione di PHP 7.x - 8.x per nuovi utenti

Argomenti

- [Per installare PHP 7.x - 8.x su un'AMI Amazon Linux 2](#)
- [Per installare PHP 7.x - 8.x su un'AMI Amazon Linux 201609](#)
- [Per installare PHP 7.x - 8.x su un'AMI SUSE Linux 15](#)
- [Per installare PHP 7.x - 8.x su un'AMI Ubuntu 22.04](#)

Per installare PHP 7.x - 8.x su un'AMI Amazon Linux 2

Note

Se necessario, sostituisci *PHP-7.x* con la versione utilizzata.

1. Avviare una nuova istanza dall'AMI.
2. Eseguì il comando seguente:

```
sudo yum install gcc-c++ zlib-devel
```

3. Installare PHP 7.x usando `amazon-linux-extras`

Con Amazon Linux 2, puoi utilizzare la Extras Library (Libreria degli extra) per installare gli aggiornamenti di applicazioni e software nelle istanze. Questi aggiornamenti di software sono definiti argomenti. Puoi installare una versione specifica di un argomento oppure omettere le informazioni sulla versione per utilizzare la versione più recente. Per ulteriori informazioni, consulta [Extras Library \(Amazon Linux 2\)](#).

Di seguito sono riportati i passaggi.

- a. Innanzitutto, verifica se `amazon-linux-extra` è già installato.

- b. Se neanche pip è installato, eseguire il seguente comando per installarlo:

```
sudo yum install -y amazon-linux-extras
```

- c. Conferma che l'argomento PHP 7.x sia disponibile nella macchina Amazon Linux 2:

```
sudo amazon-linux-extras | grep php
```

- d. Dall'output, rivedi tutti gli argomenti di PHP 7 e scegli la versione desiderata:

```
sudo amazon-linux-extras enable php7.x
```

- e. Installare il pacchetto dai repository. Esempio:

```
sudo yum clean metadata
```

```
sudo yum install php php-devel
```

4. Download del client del cluster Amazon ElastiCache .NET per ElastiCache

- Aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.

Sotto il pannello di controllo ElastiCache, vai a ElastiCache Cluster Client e quindi scegli la versione PHP7 desiderata.

- Dalla riga di comando, sostituisci PHP-7.X con la versione PHP desiderata e sostituisci ARCH con l'architettura desiderata (X86 o arm) e per PHP >= 7.4 sostituisci OpenSSL con la versione di OpenSSL desiderata (openssl1.1 oppure openssl3). Se utilizzi PHP > 7.4, rimuovi il suffisso OpenSSL.

```
wget https://elasticache-downloads.s3.amazonaws.com/ClusterClient/PHP-7.X/  
latest-64bit-<ARCH>-<OpenSSL>
```

5. Utilizzare tar -zxvf per estrarre il file scaricato.

```
tar -zxvf latest-64bit-<ARCH>-<OpenSSL>
```

6. Con autorizzazioni root, copiare il file dell'artefatto estratto amazon-elasticache-cluster-client.so in /usr/lib64/php/modules.

```
sudo mv amazon-elasticache-cluster-client.so /usr/lib64/php/modules/
```

7. Aggiungi extension=amazon-elasticache-cluster-client.so nel file/etc/php.ini

8. Se hai scaricato il client del cluster ElastiCache con PHP 7.4 o versione successiva, installa OpenSSL 1.1.x o versione successiva. Istruzioni per l'installazione per OpenSSL 1.1.1:

```
sudo yum -y update
sudo yum install -y make gcc perl-core pcre-devel wget zlib-devel
wget https://www.openssl.org/source/openssl-1.1.1c.tar.gz
tar xvf openssl-1.1.1c.tar.gz
cd openssl-1.1.1c
./config
make
sudo make install
sudo ln -s /usr/local/lib64/libssl.so.1.1 /usr/lib64/libssl.so.1.1
```

Per installare PHP 7.x - 8.x su un'AMI Amazon Linux 201609

Note

Se necessario, sostituisci *php7.x* con la versione utilizzata.

1. Avviare una nuova istanza dall'AMI. Per ulteriori informazioni su come eseguire questa operazione, consulta [Fase 1: Avvio di un'istanza](#) nella Guida utente Amazon EC2;
2. Esegui il comando seguente:

```
sudo yum install gcc-c++
```

3. Installazione di PHP

```
sudo yum install php7.x
```

4. Download del client del cluster Amazon ElastiCache .NET per ElastiCache

```
wget https://elasticache-downloads.s3.amazonaws.com/ClusterClient/PHP-7.x/
latest-64bit
```

5. Estrarre latest-64bit.

```
tar -zxvf latest-64bit
```

6. Con autorizzazione root, copiare il file dell'artefatto estratto `amazon-elasticache-cluster-client.so` in `/usr/lib64/php/7.x/modules/`.

```
sudo mv artifact/amazon-elasticache-cluster-client.so /usr/lib64/php/7.x/modules/
```

7. Crea il file `50-memcached.ini`.

```
echo "extension=amazon-elasticache-cluster-client.so" | sudo tee --append /etc/php-7.x.d/50-memcached.ini
```

8. Avviare o riavviare il server Apache.

```
sudo /etc/init.d/httpd start
```

Per installare PHP 7.x - 8.x su un'AMI SUSE Linux 15

Note

Se necessario, sostituisci `php7.x` con la versione utilizzata.

1. Avviare una nuova istanza dall'AMI.
2. Esegui il comando seguente:

```
sudo zypper refresh
sudo zypper update -y
sudo zypper install gcc
```

3. Installazione di PHP

```
sudo yum install php7.x
```

oppure

```
sudo zypper addrepo //download.opensuse.org/repositories/devel:/languages:/php/  
openSUSE_Leap_15.3/ php
```

4. Scarica il client del cluster Amazon ElastiCache e sostituisci <ARCH> con l'architettura desiderata (X86 o arm). SUSE 15 integra OpenSSL1.1, quindi per PHP >= 7.4 scegli il binario del client con OpenSSL1. Se utilizzi PHP < 7.4, rimuovi il suffisso OpenSSL.

```
wget https://elasticache-downloads.s3.amazonaws.com/ClusterClient/PHP-7.x/  
latest-64bit-<ARCH>-openssl1.1
```

5. Estrarre latest-64bit.

```
tar -zxvf latest-64bit-<ARCH>-openssl1.1
```

6. Con autorizzazione root, copiare il file dell'artefatto estratto amazon-elasticache-cluster-client.so in /usr/lib64/php7/extensions/.

```
sudo mv artifact/amazon-elasticache-cluster-client.so /usr/lib64/php7/extensions/
```

7. Inserire la riga extension=amazon-elasticache-cluster-client.so nel file /etc/php7/cli/php.ini.

```
echo "extension=amazon-elasticache-cluster-client.so" | sudo tee --append /etc/  
php7/cli/php.ini
```

8. Avviare o riavviare il server Apache.

```
sudo /etc/init.d/httpd start
```

Per installare PHP 7.x - 8.x su un'AMI Ubuntu 22.04

Note

Se necessario, sostituisci *php7.x* con la versione utilizzata.

1. Avviare una nuova istanza dall'AMI.

2. Esegui il comando seguente:

```
sudo apt-get update
sudo apt-get install gcc g++ make zlib1g zlib1g-dev
```

3. Installazione di PHP

a. Istruzioni per l'installazione per PHP 8.1:

```
sudo apt install php8.1-cli php8.1-dev
```

b. Istruzioni per l'installazione per PHP 7.4:

```
sudo apt -y install software-properties-common
sudo add-apt-repository ppa:ondrej/php
sudo apt-get update
sudo apt -y install php7.4
```

4. Scarica il client del cluster Amazon ElastiCache e sostituisci <ARCH> con l'architettura desiderata (X86 o arm). Ubuntu 22.04 integra OpenSSL3, quindi per PHP >= 7.4 scegli il binario del client con OpenSSL3. Se utilizzi PHP < 7.4, rimuovi il suffisso OpenSSL.

```
wget https://elasticache-downloads.s3.amazonaws.com/ClusterClient/PHP-7.x/
latest-64bit-<ARCH>-openssl3
```

5. Estrai latest-64bit.

```
tar -zxvf latest-64bit-<ARCH>-openssl3
```

6. Con autorizzazione root, copia il file dell'artefatto estratto `amazon-elasticache-cluster-client.so` nella directory dell'estensione php `/usr/lib/php/20190902`. Se questa estensione dir non esiste, puoi trovarla eseguendo: `php -i | grep extension_dir`
7. Inserire la riga `extension=amazon-elasticache-cluster-client.so` nel file `/etc/php/7.x/cli/php.ini`.

Installazione di PHP 5.x per nuovi utenti

Argomenti

- [Per installare PHP 5 su un'AMI Amazon Linux 2014.03 \(64 bit e 32 bit\)](#)
- [Per installare PHP 5 su un'AMI Red Hat Enterprise Linux 7.0 \(64 bit e 32 bit\)](#)
- [Per installare PHP 5 su un'AMI Ubuntu Server 14.04 LTS \(64 bit e 32 bit\)](#)
- [Per installare PHP 5 per l'AMI SUSE Linux Enterprise Server 11 \(64 bit o 32 bit\)](#)
- [Altre distribuzioni Linux](#)

Per installare PHP 5 su un'AMI Amazon Linux 2014.03 (64 bit e 32 bit)

1. Avviare un'istanza Amazon Linux (a 64 bit o 32 bit) ed eseguire l'accesso.
2. Installare le dipendenze PHP:

```
$ sudo yum install gcc-c++ php php-pear
```

3. Scaricare il pacchetto php-memcached corretto per l'istanza Amazon EC2 e la versione PHP. Per ulteriori informazioni, consulta [Download del pacchetto di installazione](#).
4. Installare php-memcached. L'URI deve essere il percorso di download per il pacchetto di installazione:

```
$ sudo pecl install <package download path>
```

Di seguito è riportato un comando di installazione di esempio per PHP 5.4, Linux a 64 bit. In questo esempio, sostituire *X.Y.Z* con il numero di versione effettivo:

```
$ sudo pecl install /home/AmazonElastiCacheClusterClient-X.Y.Z-PHP54-64bit.tgz
```

Note

Assicurati di utilizzare la versione più recente dell'artefatto di installazione.

5. Con autorizzazione root/sudo, aggiungere un nuovo file denominato memcached.ini nella directory /etc/php.d e inserire "extension=amazon-elasticache-cluster-client.so" nel file:

```
$ echo "extension=amazon-elasticache-cluster-client.so" | sudo tee --append /etc/php.d/memcached.ini
```

6. Avviare o riavviare il server Apache.

```
sudo /etc/init.d/httpd start
```

Per installare PHP 5 su un'AMI Red Hat Enterprise Linux 7.0 (64 bit e 32 bit)

1. Avviare un'istanza Red Hat Enterprise Linux (64 bit o 32 bit) ed eseguire l'accesso.
2. Installare le dipendenze PHP:

```
sudo yum install gcc-c++ php php-pear
```

3. Scaricare il pacchetto php-memcached corretto per l'istanza Amazon EC2 e la versione PHP. Per ulteriori informazioni, consulta [Download del pacchetto di installazione](#).
4. Installare php-memcached. L'URI deve essere il percorso di download per il pacchetto di installazione:

```
sudo pecl install <package download path>
```

5. Con autorizzazione root/sudo, aggiungere un nuovo file denominato memcached.ini nella directory /etc/php.d e inserire extension=amazon-elasticache-cluster-client.so nel file.

```
echo "extension=amazon-elasticache-cluster-client.so" | sudo tee --append /etc/php.d/memcached.ini
```

6. Avviare o riavviare il server Apache.

```
sudo /etc/init.d/httpd start
```

Per installare PHP 5 su un'AMI Ubuntu Server 14.04 LTS (64 bit e 32 bit)

1. Avviare un'istanza Ubuntu Linux (a 64 bit o 32 bit) ed eseguire l'accesso.
2. Installare le dipendenze PHP:

```
sudo apt-get update
```



```
sudo apt-get install gcc g++ php5 php-pear
```

3. Scaricare il pacchetto php-memcached corretto per l'istanza Amazon EC2 e la versione PHP. Per ulteriori informazioni, consulta [Download del pacchetto di installazione](#).
4. Installare php-memcached. L'URI deve essere il percorso di download per il pacchetto di installazione.

```
$ sudo pecl install <package download path>
```

Note

Questa fase di installazione consente di installare l'artefatto di compilazione `amazon-elasticache-cluster-client.so` nella directory `/usr/lib/php5/20121212*`. Verifica il percorso assoluto dell'artefatto di compilazione perché è richiesto nella fase successiva.

Se il comando precedente non funziona, occorre estrarre manualmente l'artefatto client PHP `amazon-elasticache-cluster-client.so` dal file `*.tgz` scaricato e copiarlo nella directory `/usr/lib/php5/20121212*`.

```
$ tar -xvf <package download path>
cp amazon-elasticache-cluster-client.so /usr/lib/php5/20121212/
```

5. Con autorizzazione `root/sudo`, aggiungere un nuovo file denominato `memcached.ini` nella directory `/etc/php5/cli/conf.d` e inserire `"extension=<absolute path to amazon-elasticache-cluster-client.so>"` nel file.

```
$ echo "extension=<absolute path to amazon-elasticache-cluster-client.so>" | sudo
tee --append /etc/php5/cli/conf.d/memcached.ini
```

6. Avviare o riavviare il server Apache.

```
sudo /etc/init.d/httpd start
```

Per installare PHP 5 per l'AMI SUSE Linux Enterprise Server 11 (64 bit o 32 bit)

1. Avviare un'istanza SUSE Linux (64 bit o 32 bit) ed eseguire l'accesso.
2. Installare le dipendenze PHP:

```
$ sudo zypper install gcc php53-devel
```

3. Scaricare il pacchetto php-memcached corretto per l'istanza Amazon EC2 e la versione PHP. Per ulteriori informazioni, consulta [Download del pacchetto di installazione](#).
4. Installare php-memcached. L'URI deve essere il percorso di download per il pacchetto di installazione.

```
$ sudo pecl install <package download path>
```

5. Con autorizzazione root/sudo, aggiungere un nuovo file denominato memcached.ini nella directory /etc/php5/conf.d e inserire **extension=amazon-elasticache-cluster-client.so** nel file.

```
$ echo "extension=amazon-elasticache-cluster-client.so" | sudo tee --append /etc/php5/conf.d/memcached.ini
```

6. Avviare o riavviare il server Apache.

```
sudo /etc/init.d/httpd start
```

Note

Se la Fase 5 non funziona per una delle piattaforme precedenti, verifica il percorso di installazione di `amazon-elasticache-cluster-client.so`. Specifica inoltre il percorso completo del file binario nell'estensione. Inoltre, verifica che la versione del PHP in uso sia supportata. Sono supportate le versioni da 5.3 a 5.5.

Altre distribuzioni Linux

In alcuni sistemi, in particolare CentOS7 e Red Hat Enterprise Linux (RHEL) 7.1, `libsasl2.so.3` ha sostituito `libsasl2.so.2`. In altri sistemi, quando carichi il client del cluster ElastiCache, viene eseguito un tentativo non riuscito per individuare e caricare `libsasl2.so.2`. Per risolvere questo problema, crea un collegamento simbolico a `libsasl2.so.3` in modo che quando il client tenta di caricare `libsasl2.so.2`, viene reindirizzato a `libsasl2.so.3`. Il codice seguente crea questo collegamento simbolico:

```
cd /usr/lib64
$ sudo ln libsasl2.so.3 libsasl2.so.2
```

Rimozione del client del cluster PHP

Argomenti

- [Rimozione di una versione precedente di PHP 7 o successiva](#)
- [Rimozione di una versione precedente di PHP 5](#)

Rimozione di una versione precedente di PHP 7 o successiva

Per rimuovere una versione precedente di PHP 7 o successiva

1. Rimuovi il file `amazon-elasticache-cluster-client.so` dalla directory lib PHP appropriata, come indicato in precedenza nelle istruzioni di installazione. Per informazioni sull'installazione consulta la sezione in [Per utenti con già installata l'estensione php-memcached](#).
2. Rimuovi la riga `extension=amazon-elasticache-cluster-client.so` dal file `php.ini`.
3. Avviare o riavviare il server Apache.

```
sudo /etc/init.d/httpd start
```

Rimozione di una versione precedente di PHP 5

Per rimuovere una versione precedente di PHP 5

1. Rimuovi l'estensione `php-memcached`:

```
sudo pecl uninstall __uri/AmazonElastiCacheClusterClient
```

2. Rimuovi il file `memcached.ini` aggiunto nella directory appropriata come indicato nelle fasi di installazione precedenti.

Compilazione del codice fonte per il client del cluster ElastiCache per PHP

In questa sezione viene illustrato come ottenere e compilare il codice fonte per il client del cluster ElastiCache per PHP.

Esistono due pacchetti che è necessario estrarre da GitHub e compilare; [aws-elasticache-cluster-client-libmemcached](#) e [aws-elasticache-cluster-client-memcached-for-php](#).

Argomenti

- [Compilazione della libreria libmemcached](#)
- [Compilazione del client di individuazione automatica ElastiCache Memcached per PHP](#)

Compilazione della libreria libmemcached

Librerie prerequisite

- OpenSSL 1.1.0 o successiva (a meno che il supporto TLS non sia disabilitato tramite `./configure --disable-tls`).
- SASL (libsasl2, a meno che il supporto SASL sia disabilitato tramite `./configure --disable-sasl`).

Per compilare la libreria aws-elasticache-cluster-client-libmemcached

1. Avviare un'istanza Amazon EC2
2. Installare le dipendenze di libreria.
 - Sull'AMI Amazon Linux 201509 o sull'AMI Amazon Linux 2

```
sudo yum -y update
sudo yum install gcc gcc-c++ autoconf libevent-devel make perl-core pcre-devel
wget zlib-devel
// Install OpenSSL 1.1.1
wget https://www.openssl.org/source/openssl-1.1.1c.tar.gz
tar xvf openssl-1.1.1c.tar.gz
cd openssl-1.1.1c
./config
make
sudo make install
sudo ln -s /usr/local/lib64/libssl.so.1.1 /usr/lib64/libssl.so.1.1
```

- Sull'AMI Ubuntu 14.04 (non richiesta per le versioni di Ubuntu con OpenSSL >= 1.1)

```
sudo apt-get update

sudo apt-get install libevent-dev gcc g++ make autoconf libsasl2-dev

// Install OpenSSL 1.1.1
wget https://www.openssl.org/source/openssl-1.1.1c.tar.gz
tar xvf openssl-1.1.1c.tar.gz
cd openssl-1.1.1c
./config
make
sudo make install
sudo ln -s /usr/local/lib/libssl.so.1.1 /usr/lib/x86_64-linux-gnu/libssl.so.1.1
```

3. Estrarre il repository e compilare il codice.

```
git clone https://github.com/aws-labs/aws-elasticache-cluster-client-libmemcached.git
cd aws-elasticache-cluster-client-libmemcached
touch configure.ac aclocal.m4 configure Makefile.am Makefile.in
mkdir BUILD
cd BUILD
../configure --prefix=<libmemcached-install-directory> --with-pic --disable-sasl
```

Se l'esecuzione di `../configure` non riesce a trovare `libssl` (OpenSSL Library), potrebbe essere necessario modificare la variabile di ambiente `PKG_CONFIG_PATH`:

```
PKG_CONFIG_PATH=/path/to/ssl/lib/pkgconfig ../configure --prefix=<libmemcached-install-directory> --with-pic --disable-sasl
```

In alternativa, se non utilizzi TLS, puoi disabilitarlo eseguendo:

```
make
sudo make install
../configure --prefix=<libmemcached-install-directory> --with-pic --disable-sasl --disable-tls
```

Compilazione del client di individuazione automatica ElastiCache Memcached per PHP

Nelle sezioni seguenti viene descritto come compilare il client di individuazione automatica ElastiCache Memcached

Argomenti

- [Compilazione del client Memcached ElastiCache per PHP 7 o successivo](#)
- [Compilazione del client ElastiCache Memcached per PHP 5](#)

Compilazione del client Memcached ElastiCache per PHP 7 o successivo

Replace (Sostituisci) PHP-7.x con la versione che si sta utilizzando.

Installa PHP:

```
sudo yum install -y amazon-linux-extras
sudo amazon-linux-extras enable php7.x
```

Esegui il set di comandi seguente nella directory del codice.

```
git clone https://github.com/aws-labs/aws-elasticache-cluster-client-memcached-for-php.git
cd aws-elasticache-cluster-client-memcached-for-php
phpize
mkdir BUILD
CD BUILD
../configure --with-libmemcached-dir=<libmemcached-install-directory> --disable-memcached-sasl
```

Se l'esecuzione di `../configure` non riesce a trovare libssl (OpenSSL Library), potrebbe essere necessario modificare la variabile di ambiente `PKG_CONFIG_PATH` nella directory dei file `.PC` di OpenSSL:

```
PKG_CONFIG_PATH=/path/to/ssl/lib/pkgconfig ../configure --with-libmemcached-dir=<path to libmemcached build directory> --disable-memcached-sasl
```

In alternativa, se non utilizzi TLS, puoi disabilitarlo eseguendo:

```
make
```

```
make install
../configure --with-libmemcached-dir=<path to libmemcached build directory> --disable-
memcached-sasl --disable-memcached-tls
```

Note

Puoi collegare staticamente la libreria libmemcached nel file binario PHP per consentirne la portabilità tra diverse piattaforme Linux. A questo scopo, esegui il comando seguente prima di make:

```
sed -i "s#-lmemcached#<libmemcached-install-directory>/lib/libmemcached.a -
lcrypt -lpthread -lm -lstdc++ -lsasl2#" Makefile
```

Compilazione del client ElastiCache Memcached per PHP 5

Compila `aws-elasticache-cluster-client-memcached-for-php` eseguendo i comandi seguenti nella cartella `aws-elasticache-cluster-client-memcached-for-php/`.

```
git clone https://github.com/aws-labs/aws-elasticache-cluster-client-memcached-for-
php/tree/php.git
cd aws-elasticache-cluster-client-memcached-for-php
sudo yum install zlib-devel
phpize
./configure --with-libmemcached-dir=<libmemcached-install-directory>
make
make install
```


Configurazione client ElastiCache

Un cluster ElastiCache è conforme al protocollo con Memcached. La coda, le applicazioni e gli strumenti più diffusi utilizzati oggi nell'ambiente Memcached esistente sono completamente compatibili con il servizio.

In questa sezione vengono trattate considerazioni specifiche per la connessione ai nodi di cache in ElastiCache.

Argomenti

- [Individuazione di endpoint e numeri di porta dei nodi](#)
- [Connessione per l'utilizzo di Individuazione automatica](#)
- [Nomi DNS e IP sottostante](#)

Individuazione di endpoint e numeri di porta dei nodi

Per effettuare la connessione a un nodo di cache, l'applicazione deve conoscere l'endpoint e il numero di porta per tale nodo.

Individuazione di endpoint e numeri di porta dei nodi (console)

Per determinare gli endpoint e i numeri di porta dei nodi

1. Accedi alla [console di gestione Amazon ElastiCache](#) e scegliere il motore in esecuzione sul cluster.

Viene visualizzato un elenco di tutti i cluster che eseguono il motore scelto.

2. Proseguire di seguito per il motore e la configurazione in esecuzione.
3. Scegliere il nome del cluster di interesse.
4. Individuare le colonne Port (Porta) ed Endpoint per il nodo desiderato.

Individuazione di endpoint e numeri di porta dei nodi di cache (AWS CLI)

Per determinare gli endpoint e i numeri di porta dei nodi di cache, utilizza il comando `describe-cache-clusters` con il parametro `--show-cache-node-info`.

```
aws elasticache describe-cache-clusters --show-cache-node-info
```

I nomi DNS completi e i numeri di porta si trovano nella sezione Endpoint dell'output.

Individuazione di endpoint e numeri di porta dei nodi di cache (API ElastiCache)

Per determinare gli endpoint e i numeri di porta dei nodi di cache, utilizza l'operazione `DescribeCacheClusters` con il parametro `ShowCacheNodeInfo=true`.

Example

```
https://elasticache.us-west-2.amazonaws.com /
  ?Action=DescribeCacheClusters
  &ShowCacheNodeInfo=true
  &SignatureVersion=4
  &SignatureMethod=HmacSHA256
  &Timestamp=20140421T220302Z
  &Version=2014-09-30
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256
  &X-Amz-Credential=<credential>
  &X-Amz-Date=20140421T220302Z
  &X-Amz-Expires=20140421T220302Z
  &X-Amz-Signature=<signature>
  &X-Amz-SignedHeaders=Host
```

Connessione per l'utilizzo di Individuazione automatica

Se le applicazioni utilizzano Individuazione automatica, è sufficiente conoscere l'endpoint di configurazione per il cluster, anziché i singoli endpoint per ogni nodo di cache. Per ulteriori informazioni, consulta [Identificazione automatica dei nodi nel cluster](#).

Note

Al momento, Individuazione automatica è disponibile solo per cluster di cache che eseguono Memcached.

Nomi DNS e IP sottostante

I client mantengono un elenco di server contenente gli indirizzi e i numeri di porta dei server che detengono i dati di cache. Durante l'utilizzo di ElastiCache, l'API `DescribeCacheClusters` (o l'utility a riga di comando `describe-cache-clusters`) restituisce una voce DNS completa e un numero di porta che possono essere utilizzati per l'elenco di server.

Important

È importante che le applicazioni client siano configurate per risolvere frequentemente nomi DNS di nodi di cache quando tentano di connettersi a un endpoint del nodo di cache.

Installazioni VPC

ElastiCache assicura che il nome DNS e l'indirizzo IP del nodo di cache non varino quando i nodi di cache vengono ripristinati in caso di fallimento.

Installazioni non VPC

ElastiCache assicura che il nome DNS di un nodo di cache non viene modificato quando i nodi di cache vengono ripristinati in caso di fallimento; tuttavia, l'indirizzo IP sottostante del nodo di cache può cambiare.

La maggior parte delle librerie client supportano connessioni ai nodi di cache persistenti per impostazione predefinita. Ti consigliamo di utilizzare connessioni a nodi di cache persistenti durante l'utilizzo di ElastiCache. Il caching DNS lato client si può verificare in più punti, inclusi librerie client, il runtime del linguaggio di programmazione o il sistema operativo del client. È opportuno rivedere la configurazione dell'applicazione a ogni livello per essere certi di risolvere frequentemente indirizzi IP per i nodi di cache.

Preparazione di un cluster

Di seguito troverai istruzioni per creare un cluster tramite la console ElastiCache, la AWS CLI o l'API ElastiCache.

Puoi anche creare un cluster ElastiCache mediante [AWS CloudFormation](#). Per ulteriori informazioni, consulta [AWS::ElastiCache::CacheCluster](#) nella AWS Guida per l'utente per la formazione cloud, che include una guida all'implementazione di tale approccio.

È consigliabile eseguire delle operazioni preliminari a ogni creazione di un cluster , in modo da non dover effettuare aggiornamenti o modifiche sin dai primi momenti.

Argomenti

- [Determina i tuoi requisiti](#)
- [Scelta delle dimensioni dei nodi](#)

Determina i tuoi requisiti

Preparazione

Conoscere le risposte alle seguenti domande aiuta a velocizzare la creazione del cluster:

Vuoi utilizzare un servizio ElastiCache serverless o basato su istanze?

Se desideri utilizzare la cache serverless, assicurati di aver configurato correttamente il VPC, le sottoreti e i gruppi di sicurezza. Per ulteriori dettagli, consulta [Modelli di accesso a una cache ElastiCache in un Amazon VPC](#). Se desideri utilizzare un sistema basato su istanze ElastiCache, continua a leggere.

- Che tipo di istanza dei nodi ti occorre?

Per assistenza nella scelta di un tipo di nodo dell'istanza, consulta [Scelta delle dimensioni dei nodi Memcached](#).

- Intendi avviare il cluster in un cloud privato virtuale (VPC) basato su Amazon VPC?

Important

Se intendi avviare il cluster in un VPC, prima di iniziarne la creazione devi creare un gruppo di sottoreti nello stesso VPC. Per ulteriori informazioni, consulta [Sottoreti e gruppi di sottoreti](#).

ElastiCache è progettato per essere accessibile dall'interno AWS tramite Amazon EC2. Tuttavia, se il cluster viene avviato in un VPC basato su Amazon VPC e si trova in un VPC, è possibile fornire l'accesso dall'esterno di AWS. Per ulteriori informazioni, consulta [Modelli di accesso a una cache ElastiCache in un Amazon VPC](#).

- Devi personalizzare qualche valore di parametro?

In tal caso, crea un gruppo di parametri personalizzato. Per ulteriori informazioni, consulta [Creazione di un gruppo di parametri](#).

- Devi creare un tuo gruppo di sicurezza VPC?

Per ulteriori informazioni, consulta [Security in Your VPC](#).

- Come intendi implementare la tolleranza ai guasti?

Per ulteriori informazioni, consulta [Limitazione dell'impatto degli errori](#).

Argomenti

- [Requisiti di memoria e del processore](#)
- [Configurazione del cluster Memcached](#)
- [Requisiti di dimensionamento](#)
- [Requisiti di accesso](#)
- [Requisiti delle Regioni, delle Zone di disponibilità e delle Local Zones](#)

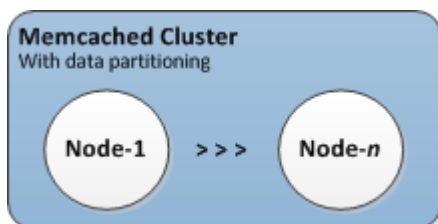
Requisiti di memoria e del processore

L'elemento costitutivo di base di Amazon ElastiCache è il nodo. Per formare i cluster, i nodi vengono configurati singolarmente o a gruppi. Quando determini il tipo di nodo da utilizzare per il cluster, prendi in considerazione la configurazione dei nodi del cluster e la quantità di dati da archiviare.

Memcached è un motore multi-thread, quindi il numero di core di un nodo influisce sulla potenza di elaborazione disponibile per il cluster.

Configurazione del cluster Memcached

ElastiCache per i cluster Memcached sono composti da 1 a 60 nodi. I dati presenti in un cluster Memcached vengono partizionati nei nodi del cluster. La tua applicazione si connette a un cluster Memcached tramite un indirizzo di rete chiamato endpoint. Ogni nodo di un cluster Memcached ha un proprio endpoint, che viene utilizzato dall'applicazione per le operazioni di lettura o scrittura sul nodo specifico. Oltre agli endpoint sul nodo primario, lo stesso cluster Memcached ha un endpoint denominato endpoint di configurazione. L'applicazione può utilizzare questo endpoint per leggere dal o scrivere sul cluster, lasciando la determinazione del nodo da cui leggere o scrivere a [Identificazione automatica dei nodi nel cluster](#).



Per ulteriori informazioni, consulta [Gestione dei cluster](#).

Requisiti di dimensionamento

Tutti i cluster sono dimensionabili tramite la creazione di un nuovo cluster con un nuovo e più grande tipo di nodo. Quando aumenti la dimensione di un cluster Memcached, il nuovo cluster inizia vuoto.

I cluster Amazon ElastiCache for Memcached possono essere scalati orizzontalmente o internamente. Per dimensionare un cluster Memcached basta aggiungere o rimuovere i nodi dal cluster. Se hai abilitato l'individuazione automatica e la tua applicazione si connette all'endpoint di configurazione del cluster, non devi apportare alcuna modifica all'applicazione quando aggiungi o rimuovi i nodi.

Per ulteriori informazioni sul tagging, consulta [Scalabilità ElastiCache per Memcached](#) in questa guida.

Requisiti di accesso

In base alla progettazione, è possibile accedere ElastiCache ai cluster Amazon dalle istanze Amazon EC2. L'accesso di rete a un ElastiCache cluster è limitato all'account che ha creato il cluster. Di conseguenza, prima di poter accedere a un cluster da un'istanza Amazon EC2, devi autorizzare l'istanza Amazon EC2 ad accedere al cluster. I passaggi per eseguire questa operazione cambiano a seconda se l'avvio è avvenuto in EC2-VPC o in EC2-Classic.

Se hai avviato il cluster in EC2-VPC devi concedere l'ingresso di rete al cluster. Se hai avviato il tuo cluster in EC2-Classic, devi concedere al gruppo di sicurezza Amazon Elastic Compute Cloud associato all'istanza l'accesso al tuo ElastiCache gruppo di sicurezza. Per istruzioni dettagliate, consultare [Accesso al cluster](#) in questa guida.

Requisiti delle Regioni, delle Zone di disponibilità e delle Local Zones

Amazon ElastiCache supporta tutte le AWS regioni. Localizzando ElastiCache i cluster in una AWS regione vicina all'applicazione, puoi ridurre la latenza. Se il cluster dispone di più nodi, posizionarli in Zone di disponibilità o in Local Zones diverse può ridurre l'effetto degli errori sul cluster.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- [Regioni e zone di disponibilità](#)
- [Zone locali](#)
- [Limitazione dell'impatto degli errori](#)

Scelta delle dimensioni dei nodi

La scelta delle dimensioni dei nodi per un cluster influisce sui costi, le prestazioni e la tolleranza ai guasti.

Scelta delle dimensioni dei nodi Memcached

I cluster Memcached possono contenere uno o più nodi, tra i quali ripartire i propri dati. Per questo motivo, le esigenze di memoria del cluster e la memoria di un nodo sono correlate, ma non corrispondenti. La necessaria capacità di memoria del cluster può contare su pochi nodi grandi o molti nodi piccoli. Inoltre, in base alle mutevoli esigenze, è possibile aggiungere nodi al cluster o rimuoverli in qualunque momento, pagando solo ciò che occorre.

La capacità di memoria totale del cluster viene calcolata moltiplicando il numero di nodi nel cluster per la capacità di RAM di ciascun nodo, al netto del carico operativo del sistema. La capacità di un nodo dipende dalla sua tipologia.

$$\text{cluster_capacity} = \text{number_of_nodes} * (\text{node_capacity} - \text{system_overhead})$$

Il numero dei suoi nodi è un fattore determinante per la disponibilità di un cluster che esegue Memcached. Il fallimento di un solo nodo può influire sulla disponibilità dell'applicazione e sul carico sul database back-end. In tal caso, ElastiCache fornisce una sostituzione per un nodo fallito e viene ripopolato. Per ridurre questo potenziale impatto sulla disponibilità, basta distribuire la memoria e la capacità di calcolo su più nodi a capacità ridotta, anziché su meno nodi ad ampia capacità.

Se, ad esempio, occorrono 35 GB di memoria cache, è possibile impostare una qualsiasi delle seguenti configurazioni:

- 11 nodi cache .t2.medium, ciascuno con 3,22 GB di memoria e 2 thread = 35,42 GB e 22 thread.
- 6 nodi cache .m4.large, ciascuno con 6,42 GB di memoria e 2 thread = 38,52 GB e 12 thread.
- 3 nodi cache .r4.large, ciascuno con 12,3 GB di memoria e 2 thread = 36,90 GB e 6 thread.
- 3 nodi cache .m4.xlarge, ciascuno con 14,28 GB di memoria e 4 thread = 42,84 GB e 12 thread.

Opzioni disponibili per i nodi a confronto

Tipo di nodo	Memoria (in GB)	Core	Costo orario*	Nodi necessari	Memoria totale (in GiB)	Core totali	Costo mensile
cache.t2.medium	3,22	2	0,068 \$	11	35,42	22	538,56 \$

Tipo di nodo	Memoria (in GB)	Core	Costo orario*	Nodi necessari	Memoria totale (in GiB)	Core totali	Costo mensile
cache.m4.large	6,42	2	0,156 \$	6	38,52	12	673,92 \$
cache.m4.xlarge	14,28	4	0,311 \$	3	42,84	12	671,76 \$
cache.m5.xlarge	12,93	4	0,311 \$	3	38,81	12	671,76 \$
cache.m6g.large	6,85	2	0,147 \$	6	41,1	12	635 \$
cache.r4.large	12,3	2	0,228 \$	3	36,9	6	492,48 \$
cache.r5.large	13,07	2	0,216 \$	3	39,22	6	466,56 \$
cache.r6g.large	13,07	2	0,205 \$	3	42,12	6	442 \$

* Costo orario per nodo dall'8 ottobre 2020.

Costo mensile al 100% di utilizzo per 30 giorni (720 ore).

Ognuna di queste opzioni offre una disponibilità di memoria simile, ma capacità e costi di elaborazione diversi. Per confrontare i costi delle varie opzioni, consulta [Prezzi di Amazon ElastiCache](#).

Nel caso dei cluster che eseguono Memcached, parte della memoria disponibile su ciascun nodo viene utilizzata per la gestione della connessione. Per ulteriori informazioni, consulta [Sovraccarico delle connessioni Memcached](#)

L'utilizzo di più nodi richiede la distribuzione delle chiavi tra gli stessi. Ogni nodo dispone del proprio endpoint. Per gestire gli endpoint con facilità, ci si può avvalere della funzione di individuazione

automatica di ElastiCache, che consente ai programmi client di rilevare automaticamente tutti i nodi in un cluster. Per ulteriori informazioni, consulta [Identificazione automatica dei nodi nel cluster](#).

In alcuni casi, è possibile non essere sicuri di quanta capacità è necessaria. In caso affermativo, per i test consigliamo di iniziare con un cache .m5.large nodo. Quindi monitora l'utilizzo della memoria, l'utilizzo della CPU e la percentuale di riscontri nella cache con le metriche ElastiCache pubblicate su Amazon CloudWatch. Per ulteriori informazioni sui parametri CloudWatch per ElastiCache, consulta [Monitoraggio dell'utilizzo con i parametri di CloudWatch](#). Per carichi di lavoro di produzione particolarmente consistenti, i nodi R5 garantiscono le prestazioni e il valore di costo della RAM migliori.

Se il cluster non garantisce l'hit rate desiderato, basta aggiungere più nodi, aumentando così la memoria totale in esso disponibile.

Se il cluster risulta vincolato dalla CPU ma, comunque, con un hit rate sufficiente, prova a configurare un nuovo cluster con un tipo di nodo che garantisca maggiore potenza di calcolo.

Creazione di un cluster

Gli esempi seguenti mostrano come creare un cluster utilizzando l'API AWS Management Console AWS CLI e ElastiCache .

Creazione di un cluster Memcached (console)

Quando utilizzi il motore Memcached, Amazon ElastiCache supporta il partizionamento orizzontale dei dati su più nodi. Memcached permette l'individuazione automatica per non dover tenere traccia degli endpoint di ciascun nodo. Memcached monitora l'endpoint di ciascun nodo, aggiornando l'elenco degli endpoint man mano che i nodi vengono aggiunti e rimossi. Per interagire con il cluster, all'applicazione è sufficiente l'endpoint di configurazione. Per ulteriori informazioni sulla funzione Individuazione automatica, consulta [Identificazione automatica dei nodi nel cluster](#).

Per creare un cluster Memcached, segui la procedura descritta in [Fase 1. Creazione di una cache](#).

Quando lo stato del cluster diventa disponibile, è possibile concedere a Amazon EC2 l'accesso al cluster, connettersi al cluster e iniziare a utilizzarlo. Per ulteriori informazioni, consultare [Accesso al cluster](#) e [Connessione manuale ai nodi di cache](#).

Important

Non appena il cluster diventa disponibile, viene addebitata ogni ora o frazione di ora in cui il cluster è attivo, anche se non viene effettivamente utilizzato. Per evitare di sostenere i costi del cluster, è necessario eliminarlo. Per informazioni, consulta [Eliminazione di un cluster](#).

Creazione di un cluster (AWS CLI)

Per creare un cluster utilizzando, usa il AWS CLI comando. `create-cache-cluster`

Important

Non appena il cluster diventa disponibile, viene addebitata ogni ora o frazione di ora in cui il cluster è attivo, anche se non viene effettivamente utilizzato. Per evitare di sostenere i costi del cluster, è necessario eliminarlo. Per informazioni, consulta [Eliminazione di un cluster](#).

Creazione di un cluster di cache Memcached (AWS CLI)

Il seguente codice CLI crea un cluster di cache Memcached con 3 nodi.

Per Linux, macOS o Unix:

```
aws elasticache create-cache-cluster \  
--cache-cluster-id my-cluster \  
--cache-node-type cache.r4.large \  
--engine memcached \  
--engine-version 1.4.24 \  
--cache-parameter-group default.memcached1.4 \  
--num-cache-nodes 3
```

Per Windows:

```
aws elasticache create-cache-cluster ^  
--cache-cluster-id my-cluster ^  
--cache-node-type cache.r4.large ^  
--engine memcached ^  
--engine-version 1.4.24 ^  
--cache-parameter-group default.memcached1.4 ^  
--num-cache-nodes 3
```

Creazione di un cluster (ElastiCache API)

Per creare un cluster utilizzando l' ElastiCache API, utilizza l'CreateCacheClusterazione.

Important

Non appena il cluster diventa disponibile, viene addebitata ogni ora o frazione di ora in cui il cluster è attivo, anche se non viene utilizzato. Per evitare di sostenere i costi del cluster, è necessario eliminarlo. Per informazioni, consulta [Eliminazione di un cluster](#).

Creazione di un cluster di cache Memcached (API) ElastiCache

Il codice seguente crea un cluster Memcached con 3 nodi (API). ElastiCache

Le interruzioni di riga vengono aggiunte per facilitare la lettura.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=CreateCacheCluster  
  &CacheClusterId=my-cluster  
  &CacheNodeType=cache.r4.large  
  &Engine=memcached  
  &NumCacheNodes=3  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150508T220302Z  
  &Version=2015-02-02  
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
  &X-Amz-Credential=<credential>  
  &X-Amz-Date=20150508T220302Z  
  &X-Amz-Expires=20150508T220302Z  
  &X-Amz-SignedHeaders=Host  
  &X-Amz-Signature=<signature>
```

Visualizzazione dei dettagli di un cluster

È possibile visualizzare informazioni dettagliate su uno o più cluster utilizzando la console , la AWS CLI o l'API ElastiCache.

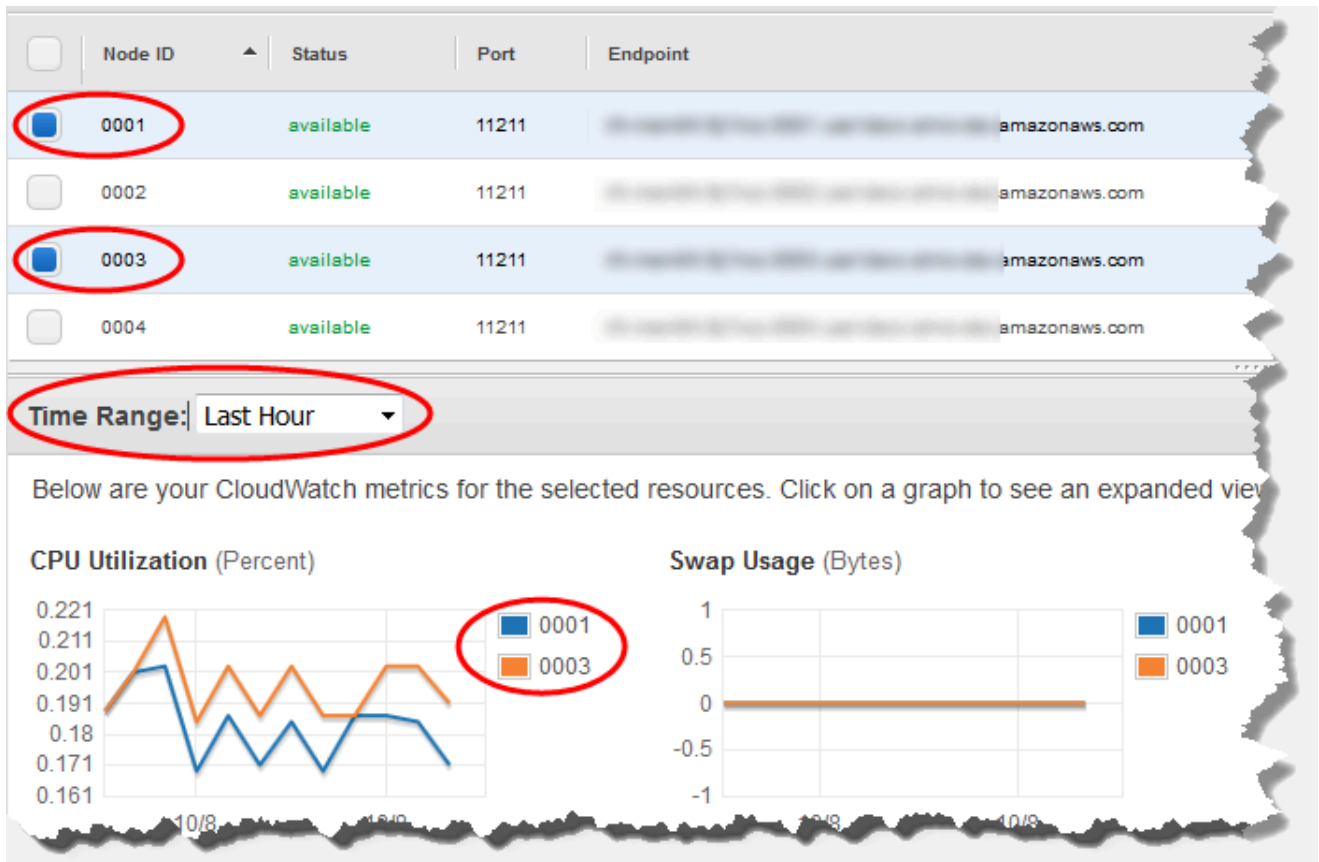
Visualizzazione dei dettagli di un cluster (console)

È possibile visualizzare i dettagli di un cluster Memcached utilizzando la console, la AWS CLI di ElastiCache o l'API ElastiCache.

La seguente procedura mostra come visualizzare i dettagli di un cluster Memcached utilizzando la console ElastiCache.

Per visualizzare i dettagli di un cluster Memcached

1. Accedi alla AWS Management Console e apri la console di Amazon ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Dall'elenco nell'angolo in alto a destra, scegliere la regione AWS a cui si è interessati.
3. Nel pannello di controllo della console ElastiCache, scegliere Memcached. Verrà così visualizzato un elenco di tutti i cluster che eseguono una qualsiasi versione di Memcached.
4. Per visualizzare i dettagli di un cluster, scegliere la casella a sinistra del nome del cluster.
5. Per visualizzare le informazioni del nodo:
 - a. Seleziona il nome del cluster.
 - b. Scegliere la scheda Nodes (Nodi).
 - c. Per visualizzare i parametri di uno o più nodi, scegliere la casella a sinistra dell'ID del nodo, quindi l'intervallo di tempo per i parametri nell'elenco Time range (Intervallo temporale). La scelta di più nodi genera grafici sovrapposti.



Parametri dell'ultima ora per due nodi Memcached

Visualizzazione dei dettagli di un cluster (AWS CLI)

Puoi visualizzare i dettagli di un cluster utilizzando il comando dell'AWS CLI `describe-cache-clusters`. Se si omette il parametro `--cache-cluster-id`, vengono restituiti i dettagli relativi a più cluster, fino a un massimo di `--max-items`. Se il parametro `--cache-cluster-id` è incluso, vengono restituiti solo i dettagli relativi al cluster specificato. Puoi limitare il numero di record restituiti con il parametro `--max-items`.

Il codice seguente consente di elencare i dettagli per `my-cluster`.

```
aws elasticache describe-cache-clusters --cache-cluster-id my-cluster
```

Il codice seguente consente di elencare i dettagli per un massimo di 25 cluster.

```
aws elasticache describe-cache-clusters --max-items 25
```

Example

Per Linux, macOS o Unix:

```
aws elasticache describe-cache-clusters \  
  --cache-cluster-id my-cluster \  
  --show-cache-node-info
```

Per Windows:

```
aws elasticache describe-cache-clusters ^  
  --cache-cluster-id my-cluster ^  
  --show-cache-node-info
```

Questa operazione genera un output simile al seguente (formato JSON):

```
{  
  "CacheClusters": [  
    {  
      "Engine": "memcached",  
      "CacheNodes": [  
        {  
          "CacheNodeId": "0001",  
          "Endpoint": {  
            "Port": 11211,  
            "Address": "my-cluster.7ef-  
example.0001.usw2.cache.amazonaws.com"  
          },  
          "CacheNodeStatus": "available",  
          "ParameterGroupStatus": "in-sync",  
          "CacheNodeCreateTime": "2016-09-21T16:28:28.973Z",  
          "CustomerAvailabilityZone": "us-west-2b"  
        },  
        {  
          "CacheNodeId": "0002",  
          "Endpoint": {  
            "Port": 11211,  
            "Address": "my-cluster.7ef-  
example.0002.usw2.cache.amazonaws.com"  
          },  
          "CacheNodeStatus": "available",  
          "ParameterGroupStatus": "in-sync",  
          "CacheNodeCreateTime": "2016-09-21T16:28:28.973Z",
```

```

        "CustomerAvailabilityZone": "us-west-2b"
    },
    {
        "CacheNodeId": "0003",
        "Endpoint": {
            "Port": 11211,
            "Address": "my-cluster.7ef-
example.0003.usw2.cache.amazonaws.com"
        },
        "CacheNodeStatus": "available",
        "ParameterGroupStatus": "in-sync",
        "CacheNodeCreateTime": "2016-09-21T16:28:28.973Z",
        "CustomerAvailabilityZone": "us-west-2b"
    }
],
"CacheParameterGroup": {
    "CacheNodeIdsToReboot": [],
    "CacheParameterGroupName": "default.memcached1.4",
    "ParameterApplyStatus": "in-sync"
},
"CacheClusterId": "my-cluster",
"PreferredAvailabilityZone": "us-west-2b",
"ConfigurationEndpoint": {
    "Port": 11211,
    "Address": "my-cluster.7ef-example.cfg.usw2.cache.amazonaws.com"
},
"CacheSecurityGroups": [],
"CacheClusterCreateTime": "2016-09-21T16:28:28.973Z",
"AutoMinorVersionUpgrade": true,
"CacheClusterStatus": "available",
"NumCacheNodes": 3,
"ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
"SecurityGroups": [
    {
        "Status": "active",
        "SecurityGroupId": "sg-dbe93fa2"
    }
],
"CacheSubnetGroupName": "default",
"EngineVersion": "1.4.24",
"PendingModifiedValues": {},
"PreferredMaintenanceWindow": "sat:09:00-sat:10:00",
"CacheNodeType": "cache.m3.medium"

```



```
    }  
  ]  
}
```

Per ulteriori informazioni, consulta l'argomento Per ElastiCache [describe-cache-clusters](#).

Visualizzazione dei dettagli di un cluster (API ElastiCache)

Puoi visualizzare i dettagli di un cluster utilizzando l'operazione DescribeCacheClusters dell'API ElastiCache. Se il parametro CacheClusterId è incluso, vengono restituiti solo i dettagli relativi al cluster specificato. Se si omette il parametro CacheClusterId, vengono restituiti i dettagli relativi a più cluster, fino a un massimo di MaxRecords (valore di default 100). Il valore dei MaxRecords non può essere minore di 20 o maggiore di 100.

Il codice seguente consente di elencare i dettagli per my-cluster.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DescribeCacheClusters  
&CacheClusterId=my-cluster  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Il codice seguente consente di elencare i dettagli per un massimo di 25 cluster.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DescribeCacheClusters  
&MaxRecords=25  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Per ulteriori informazioni, consulta l'argomento di riferimento dell'API ElastiCache [DescribeCacheClusters](#).

Modifica di un cluster ElastiCache

Oltre all'aggiunta o alla rimozione di nodi da un cluster, può capitare di dover apportare altre modifiche a un cluster esistente, ad esempio aggiungere un gruppo di sicurezza, modificare la finestra di manutenzione o un gruppo di parametri.

Consigliamo di impostare la finestra di manutenzione nel momento di utilizzo più basso. Potrebbe essere quindi necessario apportare modifiche di tanto in tanto.

Quando modifichi i parametri di un cluster, la modifica viene applicata al cluster immediatamente o dopo il riavvio del cluster. Ciò è valido se modifichi il gruppo dei parametri del cluster o un valore di parametro nel gruppo dei parametri del cluster. Per determinare quando viene applicata la modifica di un determinato parametro, consulta la colonna relativa all'applicazione delle modifiche nelle tabelle in [Parametri Memcached specifici](#). Per ulteriori informazioni sul riavvio di un cluster, consulta [Riavvio di un cluster](#).

Utilizzando il AWS Management Console

Per modificare un cluster

1. Accedere AWS Management Console e aprire la ElastiCache console all'[indirizzo https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/).
2. Dall'elenco nell'angolo in alto a destra, scegli la AWS regione in cui si trova il cluster che desideri modificare.
3. Nel riquadro di navigazione, scegliere il motore in esecuzione sul cluster da modificare.

Comparirà un elenco dei cluster che eseguono il motore selezionato.

4. Nell'elenco dei cluster, per il cluster che si desidera modificare, scegli il nome.
5. Scegliere Actions (Operazioni), quindi Modify (Modifica).

Comparirà la finestra Modifica cluster.

6. Nella finestra Modify Cluster (Modifica cluster), apportare le modifiche desiderate. Le opzioni includono:
 - Compatibilità versione del motore
 - Gruppi di sicurezza VPC
 - Gruppo di parametri
 - Finestra di manutenzione

- Argomento per la notifica SNS

La casella Apply Immediately (Applica immediatamente) vale solo per le modifiche delle versioni del motore. Per applicare immediatamente le modifiche, scegliere la casella di controllo Apply Immediately (Applica immediatamente). Se non si seleziona questa casella, le modifiche apportate alle versioni del motore vengono applicate nella finestra di manutenzione successiva. Le altre modifiche, come quelle della finestra di manutenzione, vengono applicate immediatamente.

7. Scegli Modifica.

Utilizzando il AWS CLI

È possibile modificare un cluster esistente utilizzando l' AWS CLI `modify-cache-cluster` operazione. Per modificare il valore di configurazione del cluster, specificare l'ID del cluster, il parametro da modificare e il nuovo valore del parametro. L'esempio seguente modifica la finestra di manutenzione di un cluster denominato `my-cluster` e applica immediatamente la modifica.

Important

Puoi eseguire l'upgrade alle versioni più recenti del motore. Per ulteriori informazioni in merito, consulta [Versioni e aggiornamenti del motore](#) . Tuttavia, non puoi eseguire il downgrade alle versioni precedenti del motore se non eliminando il cluster esistente e creandolo di nuovo.

Per Linux, macOS o Unix:

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-cluster \  
  --preferred-maintenance-window sun:23:00-mon:02:00
```

Per Windows:

```
aws elasticache modify-cache-cluster ^  
  --cache-cluster-id my-cluster ^  
  --preferred-maintenance-window sun:23:00-mon:02:00
```

Il parametro `--apply-immediately` si applica solo alle modifiche del tipo di nodi, della versione del motore e del numero dei nodi di un cluster. Se si desidera applicare immediatamente una di queste modifiche, utilizzare il parametro `--apply-immediately`. Se si preferisce posticipare queste modifiche alla prossima finestra di manutenzione, utilizzare il parametro `--no-apply-immediately`. Le altre modifiche, come quelle della finestra di manutenzione, vengono applicate immediatamente.

Per ulteriori informazioni, vedere l' ElastiCache argomento AWS CLI per [modify-cache-cluster](#).

Utilizzo dell' ElastiCache API

È possibile modificare un cluster esistente utilizzando l'`ModifyCacheCluster` operazione ElastiCache API. Per modificare il valore di configurazione del cluster, specificare l'ID del cluster, il parametro da modificare e il nuovo valore del parametro. L'esempio seguente modifica la finestra di manutenzione di un cluster denominato `my-cluster` e applica immediatamente la modifica.

Important

Puoi eseguire l'upgrade alle versioni più recenti del motore. Per ulteriori informazioni in merito, consulta [Versioni e aggiornamenti del motore](#) . Tuttavia, non puoi eseguire il downgrade alle versioni precedenti del motore se non eliminando il cluster esistente e creandolo di nuovo.

Le interruzioni di riga vengono aggiunte per facilitare la lettura.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ModifyCacheCluster  
&CacheClusterId=my-cluster  
&PreferredMaintenanceWindow=sun:23:00-mon:02:00  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150901T220302Z  
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
&X-Amz-Date=20150202T220302Z  
&X-Amz-SignedHeaders=Host  
&X-Amz-Expires=20150901T220302Z  
&X-Amz-Credential=<credential>  
&X-Amz-Signature=<signature>
```

Il parametro `ApplyImmediately` si applica solo alle modifiche del tipo di nodi, della versione del motore e del numero dei nodi di un cluster. Se si desidera applicare immediatamente una di queste modifiche, impostare il parametro `ApplyImmediately` su `true`. Se si preferisce posticipare queste modifiche alla prossima finestra di manutenzione, impostare il parametro `ApplyImmediately` su `false`. Le altre modifiche, come quelle della finestra di manutenzione, vengono applicate immediatamente.

Per ulteriori informazioni, consulta l'argomento di riferimento sull' ElastiCache API [ModifyCacheCluster](#).

Riavvio di un cluster

Affinché vengano applicate, alcune modifiche richiedono il riavvio del cluster. Ad esempio, per alcuni parametri la modifica del valore in un gruppo di parametri si applica solo dopo un riavvio.

Quando viene riavviato, il cluster elimina tutti i dati presenti e riavvia il motore. Nel corso di questo processo non è possibile accedere al cluster. Dato che il cluster elimina tutti i dati, quando è nuovamente disponibile è vuoto.

Si può riavviare il cluster tramite la console ElastiCache, la AWS CLI o l'API ElastiCache. Sia che si utilizzi la console ElastiCache, la AWS CLI o l'API ElastiCache, si può solo inizializzare il riavvio di un singolo cluster. Per riavviare più cluster si deve ripetere il processo o l'operazione.

Utilizzo di AWS Management Console

Puoi riavviare un cluster utilizzando la console ElastiCache.

Per riavviare un cluster (console)

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Dall'elenco nell'angolo in alto a destra, scegliere la regione AWS a cui si è interessati.
3. Nel pannello di navigazione, scegliere il motore in esecuzione sul cluster da riavviare.

Viene visualizzato un elenco dei cluster che eseguono il motore scelto.

4. Scegli il cluster da riavviare selezionando il pulsante a sinistra del nome del cluster.

Scegli Operazioni e quindi Riavvia.

Se si sceglie più di un cluster, il pulsante Reboot (Riavvia) non è attivo.

Per riavviare più cluster, ripetere le fasi da 2 a 5 per ciascun cluster da riavviare. Non occorre attendere la fine del riavvio di un cluster prima di riavviane un altro.

Per riavviare un nodo specifico, seleziona il nodo e scegliere Riavvio.

Utilizzo di AWS CLI

Per riavviare un cluster (AWS CLI), utilizzare l'operazione CLI `reboot-cache-cluster`.

Per riavviare nodi specifici nel cluster, utilizzare l'operazione `--cache-node-ids-to-reboot` per elencare cluster specifici da riavviare. Il comando seguente riavvia i nodi 0001, 0002 e 0004 del cluster `my-cluster`.

Per Linux, macOS o Unix:

```
aws elasticache reboot-cache-cluster \  
  --cache-cluster-id my-cluster \  
  --cache-node-ids-to-reboot 0001 0002 0004
```

Per Windows:

```
aws elasticache reboot-cache-cluster ^  
  --cache-cluster-id my-cluster ^  
  --cache-node-ids-to-reboot 0001 0002 0004
```

Per riavviare tutti i nodi del cluster, utilizzare il parametro `--cache-node-ids-to-reboot` ed elencare tutti gli id dei nodi del cluster. Per ulteriori informazioni, consulta [reboot-cache-cluster](#).

Utilizzo dell'API ElastiCache

Per riavviare un cluster utilizzando l'API ElastiCache, utilizzare l'operazione `RebootCacheCluster`.

Per riavviare nodi specifici nel cluster, utilizzare l'operazione `CacheNodeIdsToReboot` per elencare cluster specifici da riavviare. Il comando seguente riavvia i nodi 0001, 0002 e 0004 del cluster `my-cluster`.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=RebootCacheCluster  
&CacheClusterId=my-cluster  
&CacheNodeIdsToReboot.member.1=0001  
&CacheNodeIdsToReboot.member.2=0002  
&CacheNodeIdsToReboot.member.3=0004  
&Version=2015-02-02  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Per riavviare tutti i nodi del cluster, utilizzare il parametro `CacheNodeIdsToReboot` ed elencare tutti gli id dei nodi del cluster. Per ulteriori informazioni, consulta [RebootCacheCluster](#).

Aggiunta di nodi a un cluster

L'aggiunta di nodi a un cluster Memcached aumenta il numero delle partizioni del cluster. Quando si modifica il numero delle partizioni in un cluster, alcuni dei keyspaces devono essere rimappati per essere mappati al nodo corretto. La rimappatura dei keyspaces aumenta temporaneamente il numero di mancati riscontri nella cache sul cluster. Per ulteriori informazioni, consulta [Configurazione del client ElastiCache per un efficiente bilanciamento del carico](#).

Per aggiungere nodi al cluster si può utilizzare la console di gestione ElastiCache, la AWS CLI o l'API ElastiCache.

Utilizzo di AWS Management Console

Argomenti

- [Per aggiungere nodi a un cluster \(console\)](#)

Per aggiungere nodi a un cluster (console)

La procedura seguente può essere utilizzata per aggiungere nodi a un cluster:

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nel riquadro di navigazione, scegliere il motore in esecuzione sul cluster a cui aggiungere nodi.

Viene visualizzato un elenco dei cluster che eseguono il motore scelto.

3. Dall'elenco dei cluster, per il cluster a cui si desidera aggiungere un nodo, sceglierne il nome.
4. Scegliere Add node (Aggiungi nodo).
5. Compilare le informazioni richieste nella finestra di dialogo Add Node (Aggiungi nodo).
6. Scegliere il pulsante Apply Immediately - Yes (Applica immediatamente - Sì) per aggiungere subito il nodo o No per aggiungere il nodo nella prossima finestra di manutenzione del cluster.

Impatto delle nuove richieste di aggiunta e rimozione sulle richieste in sospeso

Scenari	Operazioni in sospeso	Nuova richiesta	Risultati
Scenario 1	Eliminazione	Eliminazione	<p>La nuova richiesta di eliminazione, in sospeso o immediata, sostituisce la richiesta di eliminazione in sospeso.</p> <p>Ad esempio, se i nodi 0001, 0003 e 0007 sono in attesa di eliminazione e viene emessa una nuova richiesta di eliminazione dei nodi 0002 e 0004, verranno eliminati solo i nodi 0002 e 0004. I nodi 0001, 0003 e 0007 non verranno eliminati.</p>
Scenario 2	Eliminazione	Crea	<p>La nuova richiesta di creazione, in sospeso o immediata, sostituisce la richiesta di eliminazione in sospeso.</p> <p>Ad esempio, se i nodi 0001, 0003 e 0007 sono in attesa di eliminazione e viene emessa una nuova richiesta di creazione di un nodo, verrà creato un nuovo nodo e i nodi 0001, 0003 e 0007 non verranno eliminati.</p>
Scenario 3	Crea	Eliminazione	<p>La nuova richiesta di eliminazione, in sospeso o immediata, sostituisce la richiesta di creazione in sospeso.</p> <p>Ad esempio, se è presente una richiesta in sospeso per creare due nodi e viene emessa una nuova richiesta per eliminare il nodo 0003, non verranno creati nuovi nodi e il nodo 0003 verrà eliminato.</p>

Scenari	Operazioni in sospenso	Nuova richiesta	Risultati
Scenario 4	Crea	Crea	<p>La nuova richiesta di creazione viene aggiunta alla richiesta di creazione in sospenso.</p> <p>Ad esempio, se è presente una richiesta in sospenso per creare due nodi e viene emessa una nuova richiesta per creare tre nodi, le nuove richieste vengono aggiunte alla richiesta in sospenso e verranno creati cinque nodi.</p> <div style="border: 1px solid #f08080; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p>⚠ Important</p> <p>Se la nuova richiesta di creazione è impostata su <code>Apply Immediately</code> - Sì, tutte le richieste di creazione vengono eseguite immediatamente. Se la nuova richiesta di creazione è impostata su <code>Apply Immediately</code> - No, tutte le richieste di creazione sono in sospenso.</p> </div>

Per determinare quali operazioni sono in sospenso, scegliere l'opzione `Description` (Descrizione) e verificare quante creazioni o eliminazioni in sospenso vengono visualizzate. Non è possibile avere sia creazioni in sospenso che eliminazioni in sospenso.

7. Scegliere il pulsante `Add` (Aggiungi).

Dopo qualche momento, nell'elenco dovrebbero comparire i nuovi nodi, con lo stato `creating` (in fase di creazione). Se non compaiono, aggiornare la pagina del browser. Quando lo stato del nodo diventa `available`, il nuovo nodo è pronto per l'uso.

Utilizzo di AWS CLI

Per aggiungere nodi a un cluster tramite l'AWS CLI, utilizzare l'operazione AWS CLI `modify-cache-cluster` con i seguenti parametri:

- `--cache-cluster-id` L'ID del cluster di cache a cui aggiungere nodi.
- `--num-cache-nodes` Il parametro `--num-cache-nodes` specifica il numero di nodi nel cluster dopo l'applicazione della modifica. Per aggiungere nodi al cluster, `--num-cache-nodes` deve essere maggiore del numero attuale di nodi nel cluster. Se il valore è inferiore al numero attuale di nodi, ElastiCache prevede il parametro `cache-node-ids-to-remove` e un elenco di nodi da rimuovere dal cluster. Per ulteriori informazioni, consulta [Utilizzo di AWS CLI](#).
- `--apply-immediately` o `--no-apply-immediately`, che specificano se aggiungere i nodi subito o alla prossima finestra di manutenzione.

Per Linux, macOS o Unix:

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-cluster \  
  --num-cache-nodes 5 \  
  --apply-immediately
```

Per Windows:

```
aws elasticache modify-cache-cluster ^  
  --cache-cluster-id my-cluster ^  
  --num-cache-nodes 5 ^  
  --apply-immediately
```

Questa operazione genera un output simile al seguente (formato JSON):

```
{  
  "CacheCluster": {  
    "Engine": "memcached",  
    "CacheParameterGroup": {  
      "CacheNodeIdsToReboot": [],  
      "CacheParameterGroupName": "default.memcached1.4",  
      "ParameterApplyStatus": "in-sync"  
    },  
    "CacheClusterId": "my-cluster",  
    "PreferredAvailabilityZone": "us-west-2b",  
    "ConfigurationEndpoint": {  
      "Port": 11211,  
      "Address": "r1lh-mem000.7alc7bf-example.cfg.usw2.cache.amazonaws.com"  
    },  
  },  
}
```

```
    "CacheSecurityGroups": [],
    "CacheClusterCreateTime": "2016-09-21T16:28:28.973Z",
    "AutoMinorVersionUpgrade": true,
    "CacheClusterStatus": "modifying",
    "NumCacheNodes": 2,
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
    "SecurityGroups": [
      {
        "Status": "active",
        "SecurityGroupId": "sg-dbe93fa2"
      }
    ],
    "CacheSubnetGroupName": "default",
    "EngineVersion": "1.4.24",
    "PendingModifiedValues": {
      "NumCacheNodes": 5
    },
    "PreferredMaintenanceWindow": "sat:09:00-sat:10:00",
    "CacheNodeType": "cache.m3.medium",
  }
}
```

Per ulteriori informazioni, consulta l'argomento della AWS CLI [modify-cache-cluster](#).

Utilizzo dell'API ElastiCache

Per aggiungere nodi a un cluster (API ElastiCache)

- Chiamare l'operazione API `ModifyCacheCluster` con i parametri seguenti:
 - `CacheClusterId` L'ID del cluster a cui aggiungere nodi.
 - `NumCacheNodes` Il parametro `NumCacheNodes` specifica il numero di nodi nel cluster dopo l'applicazione della modifica. Per aggiungere nodi al cluster, `NumCacheNodes` deve essere maggiore del numero attuale di nodi nel cluster. Se il valore è inferiore al numero attuale di nodi, ElastiCache prevede il parametro `CacheNodeIdsToRemove` con un elenco di nodi da rimuovere dal cluster (consulta [Utilizzo dell'API ElastiCache](#)).
 - `ApplyImmediately` Specifica se aggiungere i nodi subito o alla prossima finestra di manutenzione.
 - `Region` Specifica la regione AWS del cluster a cui aggiungere nodi.

L'esempio seguente mostra una chiamata per aggiungere nodi a un cluster.

Example

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=ModifyCacheCluster  
  &ApplyImmediately=true  
  &NumCacheNodes=5  
&CacheClusterId=my-cluster  
&Region=us-east-2  
  &Version=2014-12-01  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20141201T220302Z  
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
  &X-Amz-Date=20141201T220302Z  
  &X-Amz-SignedHeaders=Host  
  &X-Amz-Expires=20141201T220302Z  
  &X-Amz-Credential=<credential>  
  &X-Amz-Signature=<signature>
```

Per ulteriori informazioni, consulta l'argomento API ElastiCache [ModifyCacheCluster](#).

Rimozione di nodi da un cluster

Ogni volta che si cambia il numero di nodi in un cluster Memcached, è necessario rimappare almeno una parte del keyspace in modo da mapparlo sul nodo corretto. Per informazioni più dettagliate sul bilanciamento del carico di un cluster Memcached, consulta [Configurazione del client ElastiCache per un efficiente bilanciamento del carico](#):

Puoi eliminare un nodo da un cluster utilizzando la AWS Management Console, la AWS CLI o l'API ElastiCache.

Utilizzo di AWS Management Console

Per rimuovere nodi da un cluster (console)

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Dall'elenco nell'angolo in alto a destra, scegliere la regione AWS del cluster da cui rimuovere i nodi.
3. Nel pannello di navigazione, scegliere il motore in esecuzione sul cluster da cui rimuovere un nodo.

Viene visualizzato un elenco dei cluster che eseguono il motore scelto.

4. Dall'elenco dei cluster, scegliere il nome del cluster da cui rimuovere un nodo.

Viene visualizzato un elenco dei nodi del cluster.

5. Scegliere la casella a sinistra dell'ID del nodo da rimuovere. Utilizzando la console ElastiCache si può eliminare solo un nodo alla volta, quindi se si selezionano più nodi il pulsante Delete node (Elimina nodo) non può essere utilizzato.

Viene visualizzata la pagina Delete node (Elimina nodo) .

6. Per eliminare il nodo, completare la pagina Delete Node (Elimina nodo) e scegliere Delete Node (Elimina nodo). Per mantenere il nodo, scegliere Cancel (Annulla).

Impatto delle nuove richieste di aggiunta e rimozione sulle richieste in sospeso

Scenari	Operazioni in sospeso	Nuova richiesta	Risultati
Scenario 1	Eliminazione	Eliminazione	<p>La nuova richiesta di eliminazione, in sospeso o immediata, sostituisce la richiesta di eliminazione in sospeso.</p> <p>Ad esempio, se i nodi 0001, 0003 e 0007 sono in attesa di eliminazione e viene emessa una nuova richiesta di eliminazione dei nodi 0002 e 0004, verranno eliminati solo i nodi 0002 e 0004. I nodi 0001, 0003 e 0007 non verranno eliminati.</p>
Scenario 2	Eliminazione	Crea	<p>La nuova richiesta di creazione, in sospeso o immediata, sostituisce la richiesta di eliminazione in sospeso.</p> <p>Ad esempio, se i nodi 0001, 0003 e 0007 sono in attesa di eliminazione e viene emessa una nuova richiesta di creazione di un nodo, verrà creato un nuovo nodo e i nodi 0001, 0003 e 0007 non verranno eliminati.</p>
Scenario 3	Crea	Eliminazione	<p>La nuova richiesta di eliminazione, in sospeso o immediata, sostituisce la richiesta di creazione in sospeso.</p> <p>Ad esempio, se è presente una richiesta in sospeso per creare due nodi e viene emessa una nuova richiesta per eliminare il nodo 0003, non verranno creati nuovi nodi e il nodo 0003 verrà eliminato.</p>
Scenario 4	Crea	Crea	<p>La nuova richiesta di creazione viene aggiunta alla richiesta di creazione in sospeso.</p> <p>Ad esempio, se è presente una richiesta in sospeso per creare due nodi e viene emessa una nuova</p>

Scenari	Operazioni in sospeso	Nuova richiesta	Risultati
			<p>richiesta per creare tre nodi, le nuove richieste vengono aggiunte alla richiesta in sospeso e verranno creati cinque nodi.</p> <div data-bbox="727 430 1507 888" style="border: 1px solid #f08080; padding: 10px; margin-top: 10px;"> <p>⚠ Important</p> <p>Se la nuova richiesta di creazione è impostata su <code>Applica immediatamente - Sì</code>, tutte le richieste di creazione vengono eseguite immediatamente. Se la nuova richiesta di creazione è impostata su <code>Applica immediatamente - No</code>, tutte le richieste di creazione sono in sospeso.</p> </div>

Per determinare quali operazioni sono in sospeso, scegliere l'opzione `Description` (Descrizione) e verificare quante creazioni o eliminazioni in sospeso vengono visualizzate. Non è possibile avere sia creazioni in sospeso che eliminazioni in sospeso.

Utilizzo di AWS CLI

1. Identificare gli ID dei nodi da rimuovere. Per ulteriori informazioni, consulta [Visualizzazione dei dettagli di un cluster](#).
2. Utilizzare l'operazione CLI `modify-cache-cluster` con un elenco dei nodi da rimuovere, come nell'esempio seguente.

Per rimuovere nodi da un cluster tramite l'interfaccia a riga di comando, utilizzare il comando `modify-cache-cluster` con i seguenti parametri:

- `--cache-cluster-id` L'ID del cluster di cache da cui rimuovere i nodi.
- `--num-cache-nodes` Il parametro `--num-cache-nodes` specifica il numero di nodi nel cluster dopo l'applicazione della modifica.
- `--cache-node-ids-to-remove` Un elenco di ID dei nodi da rimuovere dal cluster.

- `--apply-immediately` o `--no-apply-immediately` Specificano se rimuovere i nodi subito o alla prossima finestra di manutenzione.
- `--region` Specifica la regione AWS del cluster da cui rimuovere i nodi.

L'esempio seguente rimuove immediatamente il nodo 0001 dal cluster my-cluster.

Per Linux, macOS o Unix:

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-cluster \  
  --num-cache-nodes 2 \  
  --cache-node-ids-to-remove 0001 \  
  --region us-east-2 \  
  --apply-immediately
```

Per Windows:

```
aws elasticache modify-cache-cluster ^  
  --cache-cluster-id my-cluster ^  
  --num-cache-nodes 2 ^  
  --cache-node-ids-to-remove 0001 ^  
  --region us-east-2 ^  
  --apply-immediately
```

Questa operazione genera un output simile al seguente (formato JSON):

```
{  
  "CacheCluster": {  
    "Engine": "memcached",  
    "CacheParameterGroup": {  
      "CacheNodeIdsToReboot": [],  
      "CacheParameterGroupName": "default.memcached1.4",  
      "ParameterApplyStatus": "in-sync"  
    },  
    "CacheClusterId": "my-cluster",  
    "PreferredAvailabilityZone": "us-east-2b",  
    "ConfigurationEndpoint": {  
      "Port": 11211,  
      "Address": "rlh-mem000.7ef-example.cfg.usw2.cache.amazonaws.com"  
    },  
  },  
}
```

```

    "CacheSecurityGroups": [],
    "CacheClusterCreateTime": "2016-09-21T16:28:28.973Z", 9dcv5r
    "AutoMinorVersionUpgrade": true,
    "CacheClusterStatus": "modifying",
    "NumCacheNodes": 3,
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
    "SecurityGroups": [
      {
        "Status": "active",
        "SecurityGroupId": "sg-dbe93fa2"
      }
    ],
    "CacheSubnetGroupName": "default",
    "EngineVersion": "1.4.24",
    "PendingModifiedValues": {
      "NumCacheNodes": 2,
      "CacheNodeIdsToRemove": [
        "0001"
      ]
    },
    "PreferredMaintenanceWindow": "sat:09:00-sat:10:00",
    "CacheNodeType": "cache.m3.medium",
  }
}

```

Per ulteriori informazioni, consulta gli argomenti AWS CLI [describe-cache-cluster](#) e [modify-cache-cluster](#).

Utilizzo dell'API ElastiCache

Per rimuovere nodi utilizzando l'API ElastiCache, chiamare l'operazione API `ModifyCacheCluster` con l'ID del cluster di cache e un elenco dei nodi da rimuovere, nel seguente modo:

- `CacheClusterId` L'ID del cluster di cache da cui rimuovere i nodi.
- `NumCacheNodes` Il parametro `NumCacheNodes` specifica il numero di nodi nel cluster dopo l'applicazione della modifica.
- `CacheNodeIdsToRemove.member.n` L'elenco degli ID dei nodi da rimuovere dal cluster.
 - `CacheNodeIdsToRemove.member.1=0004`

- `CacheNodeIdsToRemove.member.1=0005`
- `ApplyImmediately` Specifica se rimuovere i nodi subito o alla prossima finestra di manutenzione.
- `Region` Specifica la regione AWS del cluster da cui rimuovere un nodo.

L'esempio seguente rimuove immediatamente i nodi 0004 e 0005 dal cluster my-cluster.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=ModifyCacheCluster  
  &CacheClusterId=my-cluster  
  &ApplyImmediately=true  
  &CacheNodeIdsToRemove.member.1=0004  
  &CacheNodeIdsToRemove.member.2=0005  
  &NumCacheNodes=3  
  &Region us-east-2  
  &Version=2014-12-01  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20141201T220302Z  
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
  &X-Amz-Date=20141201T220302Z  
  &X-Amz-SignedHeaders=Host  
  &X-Amz-Expires=20141201T220302Z  
  &X-Amz-Credential=<credential>  
  &X-Amz-Signature=<signature>
```

Per ulteriori informazioni, consulta l'argomento API ElastiCache [ModifyCacheCluster](#).

Annullamento delle operazioni in attesa di aggiunta e di eliminazione dei nodi

Se hai scelto di non applicare subito una modifica, l'operazione è pending (in attesa) fino a che non viene eseguita nella prossima finestra di manutenzione. Si possono annullare tutte le operazioni in attesa.

Per annullare un'operazione in attesa

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Dall'elenco nell'angolo in alto a destra, scegliere la regione AWS in cui si desidera annullare un'operazione in attesa, di aggiunta, o di eliminazione di nodi.
3. Nel pannello di navigazione, scegliere il motore in esecuzione sul cluster in cui sono presenti operazioni in attesa da annullare. Viene visualizzato un elenco dei cluster che eseguono il motore scelto.
4. Nell'elenco dei cluster, scegliere il nome del cluster da modificare, non la casella alla sinistra del nome del cluster, in cui sono presenti operazioni in attesa da annullare.
5. Per determinare quali operazioni sono in sospeso, scegliere l'opzione Description (Descrizione) e verificare quante creazioni o eliminazioni in sospeso vengono visualizzate. Non è possibile avere sia creazioni in sospeso che eliminazioni in sospeso.
6. Scegliere la scheda Nodes (Nodi).
7. Per annullare tutte le operazioni in attesa, fare clic su Cancel Pending (Annulla in attesa). Viene visualizzata la finestra di dialogo Cancel Pending (Annulla in attesa).
8. Confermare la scelta di annullare tutte le operazioni in attesa selezionando il pulsante Cancel Pending (Annulla in attesa); oppure, per mantenere le operazioni, scegliere Cancel (Annulla).

Eliminazione di un cluster

Fintantoché un cluster è nello stato disponibile, ne vengono addebitati i costi, anche se non è utilizzato attivamente. Per interrompere l'addebito, elimina il cluster.

Utilizzo di AWS Management Console

La procedura seguente elimina un solo cluster dalla distribuzione. Per eliminare più cluster, ripetere la procedura per ogni cluster da eliminare. Non occorre attendere la fine dell'eliminazione di un cluster prima di avviare la procedura per eliminarne un altro.

Per eliminare un cluster

1. Accedi alla AWS Management Console e apri la console di Amazon ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nel pannello di controllo della console ElastiCache, scegliere il motore in esecuzione sul cluster da eliminare.

Viene visualizzato un elenco di tutti i cluster che eseguono quel motore.

3. Per scegliere il cluster da eliminare, seleziona il nome del cluster nell'elenco di cluster.

Important

Puoi eliminare un solo cluster alla volta dalla console ElastiCache. La selezione di più cluster disabilita l'operazione di eliminazione.

4. In Actions (Azioni), scegliere Delete (Elimina).
5. Nella schermata di conferma Delete Cluster (Elimina cluster), scegliere Delete (Elimina) per eliminare il cluster o scegliere Cancel (Annulla) per mantenerlo.

Se si sceglie Delete (Elimina), lo stato del cluster diventa in fase di eliminazione.

Non appena il cluster viene rimosso dall'elenco di cluster, non ti verranno più addebitati costi.

Utilizzo di AWS CLI

Il seguente codice elimina il cluster di cache `my-cluster`.

```
aws elasticache delete-cache-cluster --cache-cluster-id my-cluster
```

L'operazione CLI `delete-cache-cluster` elimina un solo cluster di cache. Per eliminare più cluster di cache, chiamare l'operazione `delete-cache-cluster` per ogni cluster da eliminare. Non occorre attendere la fine dell'eliminazione di un cluster di cache prima di eliminarne un altro.

Per Linux, macOS o Unix:

```
aws elasticache delete-cache-cluster \  
  --cache-cluster-id my-cluster \  
  --region us-east-2
```

Per Windows:

```
aws elasticache delete-cache-cluster ^  
  --cache-cluster-id my-cluster ^  
  --region us-east-2
```

Per ulteriori informazioni, consulta l'argomento AWS CLI per ElastiCache [delete-cache-cluster](#).

Utilizzo dell'API ElastiCache

Il seguente codice elimina il cluster `my-cluster`.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DeleteCacheCluster  
&CacheClusterId=my-cluster  
&Region us-east-2  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T220302Z  
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
&X-Amz-Date=20150202T220302Z  
&X-Amz-SignedHeaders=Host  
&X-Amz-Expires=20150202T220302Z  
&X-Amz-Credential=<credential>  
&X-Amz-Signature=<signature>
```

L'operazione API `DeleteCacheCluster` elimina un solo cluster di cache. Per eliminare più cluster di cache, chiamare l'operazione `DeleteCacheCluster` per ogni cluster da eliminare. Non occorre attendere la fine dell'eliminazione di un cluster di cache prima di eliminarne un altro.

Per ulteriori informazioni, consulta l'argomento di riferimento dell'API ElastiCache [DeleteCacheCluster](#).

Accesso al cluster

Le istanze Amazon ElastiCache sono progettate in modo che vi si possa accedere tramite un'istanza Amazon EC2.

Se hai lanciato l'istanza ElastiCache in un Amazon Virtual Private Cloud (Amazon VPC), puoi accedere all'istanza ElastiCache da un'istanza Amazon EC2 nello stesso Amazon VPC. Oppure, utilizzando un peering VPC, puoi accedere all'istanza ElastiCache da un'istanza Amazon EC2 di un Amazon VPC diverso.

Se hai avviato l'istanza ElastiCache in EC2 Classic, consenti all'istanza EC2 di accedere al cluster concedendo al gruppo di sicurezza Amazon EC2 associato al cluster l'accesso al gruppo di sicurezza cache. Per impostazione predefinita, l'accesso a un cluster è riservato all'account che lo ha avviato.

Argomenti

- [Concedi l'accesso al cluster](#)

Concedi l'accesso al cluster

Il cluster è stato avviato in EC2-VPC.

Se hai lanciato il cluster in un Amazon Virtual Private Cloud (Amazon VPC), puoi connetterti al cluster ElastiCache solo da un'istanza Amazon EC2 in esecuzione nello stesso Amazon VPC. In questo caso sarà necessario concedere al cluster l'ingresso di rete.


Note

Se si utilizza Local Zones, assicurarsi di averla abilitata. Per ulteriori informazioni, consulta [Abilitazione delle zone locali](#). In questo modo, il VPC viene esteso a tale zona locale e il VPC tratterà la sottorete come qualsiasi sottorete in qualsiasi altra Zona di Disponibilità e gateway rilevanti, tabelle di instradamento e altre considerazioni sui gruppi di sicurezza.

Per concedere l'ingresso di rete a un cluster da un gruppo di sicurezza Amazon VPC

1. Accedi a AWS Management Console e apri la console Amazon EC2 all'indirizzo <https://console.aws.amazon.com/ec2/>.
2. Nel riquadro di navigazione, in Rete e sicurezza, scegli Gruppi di sicurezza.

3. Dall'elenco dei gruppi di sicurezza, scegli il gruppo di sicurezza per il VPC Amazon. A meno che il gruppo di sicurezza non sia stato creato per l'utilizzo di ElastiCache, questo gruppo di sicurezza verrà denominato di default.
4. Scegli la scheda In entrata, quindi procedi come segue:
 - a. Scegli Modifica.
 - b. Scegli Aggiungi regola.
 - c. Nella colonna Tipo, scegli Regola TCP personalizzata.
 - d. Nella casella Intervallo porta, digita il numero di porta per il nodo del cluster. Questo numero deve essere lo stesso specificato durante l'avvio del cluster. La porta predefinita per Memcached è **11211**.
 - e. Nella casella Fonte, scegli Ovunque che ha l'intervallo di porta (0.0.0.0/0) in modo che qualunque istanza Amazon EC2 avviata in Amazon VPC possa connettersi ai nodi ElastiCache.

 Important

Aprire il cluster ElastiCache nell'intervallo di porte 0.0.0.0/0 non espone il cluster a Internet perché sprovvisto di indirizzi IP pubblici e quindi non vi si può accedere dall'esterno del VPC. Tuttavia, il gruppo di sicurezza di default può essere applicato ad altre istanze Amazon EC2 nell'account del cliente, che possono avere un indirizzo IP pubblico. Se eseguono qualche attività sulla porta predefinita, il servizio può essere esposto involontariamente. Consigliamo quindi di creare un gruppo di sicurezza VPC che verrà utilizzato esclusivamente da ElastiCache. Per ulteriori informazioni, consulta [Gruppi di sicurezza personalizzati](#).

- f. Seleziona Salva.

Quando avvii un'istanza Amazon EC2 nel VPC Amazon, questa sarà in grado di connettersi al cluster ElastiCache.

Accesso alle risorse ElastiCache dall'esterno AWS

Amazon ElastiCache è un servizio AWS che fornisce un archivio di tipo chiave-valore in memoria basato sul cloud. Il servizio è progettato per potervi accedere esclusivamente da AWS. Se il cluster ElastiCache è ospitato in un VPC, puoi tuttavia utilizzare un'istanza Network Address Translation (NAT) per fornire accesso dall'esterno.

Requisiti

Per poter accedere alle risorse ElastiCache dall'esterno di AWS è necessario soddisfare i seguenti requisiti:

- Il cluster deve risiedere all'interno di un VPC e deve essere possibile accedervi tramite un'istanza NAT (Network Address Translation). Non ci sono eccezioni a questo requisito.
- L'istanza NAT deve essere avviata nello stesso VPC del cluster.
- L'istanza NAT deve essere avviata nella sottorete pubblica separata dal cluster.
- Un indirizzo IP elastico (EIP) deve essere associato all'istanza NAT. La funzionalità di inoltro della porta iptables viene utilizzata per inoltrare una porta nell'istanza NAT verso la porta del nodo di cache all'interno del VPC.

Considerazioni

Quando si accede alle risorse ElastiCache dall'esterno di ElastiCache, è necessario tenere presenti le seguenti considerazioni.

- I client si connettono all'EIP e alla porta della cache dell'istanza NAT. L'inoltro della porta sull'istanza NAT inoltra il traffico sul nodo cluster di cache appropriato.
- Se un nodo cluster viene aggiunto o sostituito, le regole iptables devono essere aggiornate per riflettere questo cambiamento.

Limitazioni

Questo approccio deve essere utilizzato esclusivamente a scopo di sviluppo e test. Non è consigliato per essere utilizzato per la produzione a causa dei limiti seguenti:

- L'istanza NAT agisce come proxy tra i client e più cluster. L'aggiunta di un proxy ha impatto sulle prestazioni del cluster di cache. L'impatto aumenta con il numero di cluster di cache a cui accedi tramite l'istanza NAT.

- Il traffico dai client verso l'istanza NAT non è crittografato. Devi pertanto evitare l'invio di dati sensibili tramite l'istanza NAT.
- L'istanza NAT aggiunge il sovraccarico della gestione di un'altra istanza.
- L'istanza NAT viene utilizzata come singolo punto di errore. Per informazioni su come configurare l'elevata disponibilità per NAT su VPC, consulta [Elevata disponibilità per istanze NAT Amazon VPC: un esempio](#).

Come accedere alle risorse ElastiCache dall'esterno di AWS

La procedura seguente mostra come eseguire la connessione alle risorse ElastiCache tramite un'istanza NAT.

Questi passaggi presuppongono quanto segue:

- `iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 6380 -j DNAT --to 10.0.1.231:6379`
- `iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 6381 -j DNAT --to 10.0.1.232:6379`

Quindi, è necessario NAT nella direzione opposta:

```
iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to-source 10.0.0.55
```

È inoltre necessario abilitare l'inoltro IP, che è disabilitato per impostazione predefinita:

```
sudo sed -i 's/net.ipv4.ip_forward=0/net.ipv4.ip_forward=1/g' /etc/sysctl.conf  
sudo sysctl --system
```

- Stai accedendo a un cluster Memcached con:
 - Indirizzo IP: 10.0.1.230
 - Porta Memcached predefinita: 11211
 - Gruppo di sicurezza: *10\0\0\55*
- L'indirizzo IP del client affidabile è 198.51.100.27.
- L'indirizzo IP elastico (EIP) dell'istanza NAT è 203.0.113.73.
- Il gruppo di sicurezza dell'istanza NAT è sg-ce56b7a9.

Per connettersi alle risorse ElastiCache tramite un'istanza NAT

1. Creare un'istanza NAT nello stesso VPC del cluster di cache, ma in una sottorete pubblica.

Per impostazione predefinita, la creazione guidata di VPC avvia un tipo di nodo cache.m1.small. È necessario selezionare una dimensione di nodo in base alle proprie esigenze: È necessario utilizzare l'AMI NAT EC2 per poter accedere a ElastiCache dall'esterno di AWS.

Per informazioni sulla creazione di un'istanza NAT, consulta [Istanze NAT](#) nella Guida per l'utente di AWS VPC.

2. Creare regole del gruppo di sicurezza per il cluster di cache e l'istanza NAT.

Il gruppo di sicurezza per le istanze NAT e l'istanza del cluster devono essere associati alle seguenti regole:

- Due regole in entrata
 - Una per consentire le connessioni TCP dai client affidabili a ciascuna porta della cache inoltrata dall'istanza NAT (11211 - 11213).
 - Un'altra per consentire l'accesso SSH ai client affidabili.

Gruppo di sicurezza per le istanze NAT - Regole in entrata

Type	Protocollo	Intervallo porte	Origine
Regola TCP personalizzata	TCP	11211-11213	198.51.100.27/32
SSH	TCP	22	198.51.100.27/32

- Una regola in uscita per consentire le connessioni TCP alla porta della cache (11211).

Gruppo di sicurezza per le istanze NAT - Regola in uscita

Type	Protocollo	Intervallo porte	Destinazione
Regola TCP personalizzata	TCP	11211	sg-ce56b7a9 (gruppo di sicurezza per le istanze del cluster)

- Una regola in entrata per il gruppo di sicurezza del cluster che consente le connessioni TCP dall'istanza NAT alla porta della cache (11211).

Gruppo di sicurezza per le istanze del cluster - Regola in entrata

Type	Protocollo	Intervallo porte	Origine
Regola TCP personalizzata	TCP	11211	sg-bd56b7 da (gruppo di sicurezza NAT)

3. Convalidare le regole.

- Confermare che il client affidabile possa effettuare la connessione SSH all'istanza NAT.
- Confermare che il client affidabile possa effettuare la connessione al cluster dall'istanza NAT.

4. Aggiungere una regola iptables all'istanza NAT.

È necessario aggiungere una regola iptables alla tabella NAT per ogni nodo nel cluster per inoltrare la porta della cache dall'istanza NAT al nodo del cluster. Di seguito è riportato un possibile esempio:

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 11211 -j DNAT --to 10.0.1.230:11211
```

Il numero di porta deve essere univoco per ciascun nodo nel cluster. Ad esempio, se si utilizza un cluster Memcached a tre nodi tramite le porte 11211 - 11213, le regole avrebbero un aspetto simile al seguente:

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 11211 -j DNAT --to 10.0.1.230:11211
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 11212 -j DNAT --to 10.0.1.231:11211
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 11213 -j DNAT --to 10.0.1.232:11211
```

5. Confermare che il client affidabile possa effettuare la connessione al cluster.

Il client affidabile deve connettersi all'EIP associato all'istanza NAT e alla porta del cluster corrispondente al nodo del cluster appropriato. Ad esempio, la stringa di connessione per PHP potrebbe apparire come segue:

```
$memcached->connect( '203.0.113.73', 11211 );  
$memcached->connect( '203.0.113.73', 11212 );  
$memcached->connect( '203.0.113.73', 11213 );
```

È anche possibile utilizzare un client telnet per verificare la connessione. Per esempio:

```
telnet 203.0.113.73 11211  
telnet 203.0.113.73 11212  
telnet 203.0.113.73 11213
```

6. Salvare la configurazione iptables.

Salvare le regole dopo averle sottoposte a test e a verifica. Se si sta utilizzando una distribuzione Linux basata su Redhat (ad esempio Amazon Linux), eseguire questo comando:

```
service iptables save
```

Argomenti correlati

Negli argomenti seguenti vengono fornite informazioni aggiuntive.

- [Modelli di accesso a una cache ElastiCache in un Amazon VPC](#)
- [Accesso a una cache ElastiCache da un'applicazione in esecuzione nel data center di un cliente](#)
- [Istanze NAT](#)
- [Configurazione di client ElastiCache](#)
- [Elevata disponibilità per istanze NAT Amazon VPC: un esempio](#)

Individuazione degli endpoint di connessione

Le applicazioni utilizzano gli endpoint per connettersi ai cluster. Un endpoint corrisponde all'indirizzo univoco di un nodo o di un cluster.

Quali endpoint utilizzare

Per la cache ElastiCache serverless con Memcached, è sufficiente acquisire dalla console il DNS e la porta dell'endpoint del cluster.

Dalla AWS CLI, usa il comando `describe-serverless-caches` per acquisire le informazioni sull'endpoint.

Linux

```
aws elasticache describe-serverless-caches --serverless-cache-name CacheName
```

Windows

```
aws elasticache describe-serverless-caches --serverless-cache-name CacheName
```

L'output dell'operazione precedente è simile al seguente (in formato JSON):

```
{
  "ServerlessCaches": [
    {
      "ServerlessCacheName": "serverless-memcached",
      "Description": "test",
      "CreateTime": 1697659642.136,
      "Status": "available",
      "Engine": "memcached",
      "MajorEngineVersion": "1.6",
      "FullEngineVersion": "21",
      "SecurityGroupIds": [
        "sg-083eda453e1e51310"
      ],
      "Endpoint": {
        "Address": "serverless-memcached-01.amazonaws.com",
        "Port": 11211
      },
      "ARN": "<the ARN>",
      "SubnetIds": [
        "subnet-0cf759df15bd4dc65",
        "subnet-09e1307e8f1560d17"
      ],
      "SnapshotRetentionLimit": 0,
      "DailySnapshotTime": "03:00"
    }
  ]
}
```

Per un'istanza basata su cluster Memcached, se usi l'individuazione automatica puoi utilizzare l'endpoint di configurazione del cluster per impostare il client Memcached. È necessario quindi utilizzare un client che supporti la funzione di individuazione automatica Auto Discovery.

In caso contrario, occorre configurare il client in modo che si avvalga degli endpoint di nodo individuali per le operazioni di lettura e scrittura. È necessario, inoltre, tenere traccia dei nodi eventualmente aggiunti o rimossi.

Nelle sezioni seguenti è illustrato, fase per fase, il processo di individuazione degli endpoint necessari per il motore in esecuzione.

Individuazione degli endpoint di un cluster (console)

Tutti gli endpoint Memcached sono sia di lettura che di scrittura. Per connettersi ai nodi in un cluster Memcached, l'applicazione può utilizzare gli endpoint di ciascun nodo o l'endpoint di configurazione del cluster, insieme all'individuazione automatica. Per avvalersi dell'individuazione automatica è necessario utilizzare un client che la supporti.

Con l'individuazione automatica, l'applicazione client si connette al cluster Memcached tramite l'endpoint di configurazione. Nel momento in cui si ridimensiona il cluster aggiungendo o rimuovendo nodi, l'applicazione viene automaticamente e immediatamente a "conoscenza" di tutti i nodi del cluster ed è in grado di connettersi a essi. Senza l'individuazione automatica, l'applicazione deve fare da sé o spetta all'utente aggiornare manualmente gli endpoint nell'applicazione, per ogni aggiunta o rimozione di un nodo.

Per copiare un endpoint, scegli l'icona di copia direttamente davanti all'indirizzo dell'endpoint. Per informazioni sull'utilizzo dell'endpoint per la connessione a un nodo, consulta [Connessione ai nodi](#).

Gli endpoint di configurazione e dei nodi sono molto simili. Le differenze sono riportate di seguito in grassetto.

```
myclustername.xxxxxx.cfg.usw2.cache.amazonaws.com:port # configuration endpoint  
contains "cfg"  
myclustername.xxxxxx.0001.usw2.cache.amazonaws.com:port # node endpoint for node 0001
```

Important

Chi sceglie di impostare un CNAME come endpoint di configurazione Memcached, affinché il client dell'individuazione automatica lo riconosca come tale, deve includere `.cfg.` nel CNAME stesso.

Individuazione degli endpoint (AWS CLI)

Per trovare gli endpoint di nodi e cluster, è possibile avvalersi di AWS CLI per Amazon ElastiCache.

Argomenti

- [Individuazione degli endpoint per nodi e cluster \(AWS CLI\)](#)

Individuazione degli endpoint per nodi e cluster (AWS CLI)

AWS CLI, con il comando `describe-cache-clusters`, consente di individuare gli endpoint di un cluster con i relativi nodi. Per i cluster Memcached, il comando restituisce l'endpoint di configurazione. Se si include il parametro facoltativo `--show-cache-node-info`, il comando restituisce, inoltre, gli endpoint dei vari nodi nel cluster.

Example

Il seguente comando recupera l'endpoint di configurazione (`ConfigurationEndpoint`) e gli endpoint dei singoli nodi (`Endpoint`) del cluster Memcached `mycluster`.

Per Linux, macOS o Unix:

```
aws elasticache describe-cache-clusters \  
  --cache-cluster-id mycluster \  
  --show-cache-node-info
```

Per Windows:

```
aws elasticache describe-cache-clusters ^  
  --cache-cluster-id mycluster ^  
  --show-cache-node-info
```

L'output dell'operazione sopra indicata dovrebbe essere simile a questo (in formato JSON).

```
{  
  "CacheClusters": [  
    {  
      "Engine": "memcached",  
      "CacheNodes": [  
        {  
          "CacheNodeId": "0001",  
          "Endpoint": {
```

```
    "Port": 11211,
    "Address": "mycluster.amazonaws.com"
  },
  "CacheNodeStatus": "available",
  "ParameterGroupStatus": "in-sync",
  "CacheNodeCreateTime": "2016-09-22T21:30:29.967Z",
  "CustomerAvailabilityZone": "us-west-2b"
},
{
  "CacheNodeId": "0002",
  "Endpoint": {
    "Port": 11211,
    "Address": "mycluster.amazonaws.com"
  },
  "CacheNodeStatus": "available",
  "ParameterGroupStatus": "in-sync",
  "CacheNodeCreateTime": "2016-09-22T21:30:29.967Z",
  "CustomerAvailabilityZone": "us-west-2b"
},
{
  "CacheNodeId": "0003",
  "Endpoint": {
    "Port": 11211,
    "Address": "mycluster.amazonaws.com"
  },
  "CacheNodeStatus": "available",
  "ParameterGroupStatus": "in-sync",
  "CacheNodeCreateTime": "2016-09-22T21:30:29.967Z",
  "CustomerAvailabilityZone": "us-west-2b"
}
],
"CacheParameterGroup": {
  "CacheNodeIdsToReboot": [],
  "CacheParameterGroupName": "default.memcached1.4",
  "ParameterApplyStatus": "in-sync"
},
"CacheClusterId": "mycluster",
"PreferredAvailabilityZone": "us-west-2b",
"ConfigurationEndpoint": {
  "Port": 11211,
  "Address": "mycluster.amazonaws.com"
},
"CacheSecurityGroups": [],
"CacheClusterCreateTime": "2016-09-22T21:30:29.967Z",
```

```
    "AutoMinorVersionUpgrade": true,  
    "CacheClusterStatus": "available",  
    "NumCacheNodes": 3,  
    "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/  
home#client-download:",  
    "CacheSubnetGroupName": "default",  
    "EngineVersion": "1.4.24",  
    "PendingModifiedValues": {},  
    "PreferredMaintenanceWindow": "mon:09:00-mon:10:00",  
    "CacheNodeType": "cache.m4.large",  
  
  }  
]  
}
```

Important

Chi sceglie di impostare un CNAME come endpoint di configurazione Memcached, affinché il client dell'individuazione automatica lo riconosca come tale, deve includere `.cfg.` nel CNAME stesso. Ad esempio, `mycluster.cfg.local` nel file `php.ini` per il parametro `session.save_path`.

Per ulteriori informazioni, consulta l'argomento [describe-cache-clusters](#).

Ricerca degli endpoint (API ElastiCache)

Per trovare gli endpoint di nodi e cluster, è possibile avvalersi dell'API di Amazon ElastiCache.

Argomenti

- [Individuazione degli endpoint di nodi e cluster \(API ElastiCache\)](#)

Individuazione degli endpoint di nodi e cluster (API ElastiCache)

L'API di ElastiCache, con l'operazione `DescribeCacheClusters`, consente di individuare gli endpoint di un cluster con i relativi nodi. Per i cluster Memcached, il comando restituisce l'endpoint di configurazione. Se si include il parametro facoltativo `ShowCacheNodeInfo`, l'operazione restituisce, inoltre, gli endpoint dei vari nodi nel cluster.

Example

Il seguente comando recupera l'endpoint di configurazione (`ConfigurationEndpoint`) e gli endpoint dei singoli nodi (`Endpoint`) del cluster Memcached `mycluster`.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=DescribeCacheClusters  
  &CacheClusterId=mycluster  
  &ShowCacheNodeInfo=true  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &Version=2015-02-02  
  &X-Amz-Credential=<credential>
```

Important

Chi sceglie di impostare un CNAME come endpoint di configurazione Memcached, affinché il client dell'individuazione automatica lo riconosca come tale, deve includere `.cfg.` nel CNAME stesso. Ad esempio, `mycluster.cfg.local` nel file `php.ini` per il parametro `session.save_path`.

Gestione di nodi

Un nodo è l'elemento base di una distribuzione Amazon ElastiCache. Corrisponde a un blocco di dimensioni fisse di RAM collegata alla rete. Ogni nodo esegue il motore che è stato scelto al momento della creazione e dell'ultima modifica del cluster. Ogni nodo dispone del proprio nome Domain Name Service (DNS) e porta. Sono supportati più tipi di nodi ElastiCache, ciascuno con quantità variabili di memoria associata e potenza di calcolo.

Per una discussione dettagliata sulle dimensioni del nodo da utilizzare, consultare [Scelta delle dimensioni dei nodi Memcached](#).

Argomenti

- [Visualizzazione dello stato ElastiCache del nodo](#)
- [Connessione ai nodi](#)
- [Tipi di nodi supportati](#)
- [Sostituzione dei nodi](#)
- [Nodi riservati ElastiCache](#)
- [Migrazione dei nodi della generazione precedente](#)

Alcune importanti operazioni che coinvolgono i nodi sono le seguenti:

- [Aggiunta di nodi a un cluster](#)
- [Rimozione di nodi da un cluster](#)
- [Scalabilità ElastiCache per Memcached](#)
- [Individuazione degli endpoint di connessione](#)

Visualizzazione dello stato ElastiCache del nodo

Utilizzando la [ElastiCache console](#), puoi accedere rapidamente allo stato del tuo ElastiCache nodo. Lo stato di un ElastiCache nodo indica lo stato del nodo. Puoi utilizzare le seguenti procedure per visualizzare lo stato del ElastiCache nodo nella ElastiCache console Amazon, il AWS CLI comando o il funzionamento dell'API.

I possibili valori di stato per ElastiCache i nodi sono riportati nella tabella seguente. Questa tabella mostra anche se ti verrà addebitato il costo del ElastiCache nodo.

Type	Fatturato	Descrizione
available	Fatturato	Il ElastiCache nodo è integro e disponibile.
creating	Non fatturata	Il ElastiCache nodo è in fase di creazione. Non è possibile accedere al nodo mentre è in fase di creazione.
deleting	Non fatturata	Il ElastiCache nodo viene eliminato.
modifying	Fatturato	Il ElastiCache nodo viene modificato a causa di una richiesta del cliente di modificare il nodo.
updating	Fatturato	<p>Uno stato di aggiornamento indica che una o più delle seguenti condizioni sono vere per il ElastiCache nodo Amazon:</p> <ul style="list-style-type: none">• Il ElastiCache nodo viene patchato come parte dell'aggiornamento del servizio. Per ulteriori informazioni sugli aggiornamenti del servizio, consulta la pagina di aiuto di Amazon ElastiCache Managed Maintenance and Service Updates.• I gruppi di sicurezza VPC si stanno aggiornando per il ElastiCache Cluster.

Type	Fatturato	Descrizione
		<ul style="list-style-type: none">• Il ElastiCache cluster viene scalato verso l'alto o verso il basso.• Le configurazioni di consegna dei log sono in fase di modifica per il Cluster. ElastiCache• Un'operazione di eliminazione per il ElastiCache nodo è in sospeso.• La password ElastiCache per Redis viene aggiornata/ruotata utilizzando. AWS Secrets Manager
rebooting cache cluster nodes	Fatturato	Il ElastiCache nodo viene riavviato a causa di una richiesta del cliente o di un ElastiCache processo Amazon che richiede il riavvio del nodo.

Type	Fatturato	Descrizione
<code>incompatible_parameters</code>	Non fatturata	<p>Amazon non ElastiCache può avviare il nodo perché i parametri specificati nel gruppo di parametri del nodo non sono compatibili con il nodo. Annulla le modifiche apportate ai parametri o rendi tali parametri compatibili con il nodo per ottenere di nuovo l'accesso al nodo. Per ulteriori informazioni sui parametri incompatibili, consulta l'elenco degli eventi per il ElastiCache nodo.</p>
<code>incompatible_network</code>	Non fatturata	<p>Uno stato di rete incompatibile indica che una o più delle seguenti condizioni sono vere per il nodo Amazon: ElastiCache</p> <ul style="list-style-type: none">• Non ci sono indirizzi IP disponibili nella sottorete in cui è stato avviato il ElastiCache nodo.• La sottorete menzionata nel gruppo di ElastiCache sottoreti non esiste più in Amazon Virtual Private Cloud (Amazon VPC).

Type	Fatturato	Descrizione
restore_failed	Non fatturata	<p>Uno stato di ripristino non riuscito indica che per il nodo Amazon ElastiCache si verifica una delle seguenti condizioni:</p> <ul style="list-style-type: none">• Le sostituzioni del nodo non sono riuscite a causa della capacità insufficiente dell'istanza registrata a ripetutamente. Ciò si verifica in genere quando si eseguono nodi di generazione precedente che lo sono. end-of-life Tuttavia, potrebbe verificarsi anche in caso di sostituzione dei nodi della generazione attuale quando AWS non si dispone di una capacità on-demand sufficiente per soddisfare la richiesta nella zona di disponibilità specificata. Per ulteriori informazioni sulla correzione o la rimozione di questi nodi, vedere Migrazione dei nodi della generazione precedente e.• Lo snapshot RDB specifico non è stato ripristinato.• L' AWS account per il ElastiCache cluster è stato sospeso.

Type	Fatturato	Descrizione
		<ul style="list-style-type: none">Il nodo non è riuscito e non può essere ripristinato.
snapshotting	Fatturato	ElastiCache sta creando un'istantanea del nodo ElastiCache for Redis.

Visualizzazione dello stato ElastiCache del nodo con la console

Per visualizzare lo stato di un ElastiCache nodo con la console:

- Accedi AWS Management Console e apri la ElastiCache console Amazon all'[indirizzo https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/).
- Nel pannello di navigazione, scegli Redis Clusters o Memcached Clusters. Viene visualizzata la pagina Cache con l'elenco dei nodi. ElastiCache Per ogni nodo viene visualizzato il valore dello stato.
- È quindi possibile accedere alla scheda Service Updates relativa alla cache per visualizzare l'elenco degli aggiornamenti del servizio applicabili alla cache.

Visualizzazione dello stato del ElastiCache nodo con AWS CLI

Per visualizzare il ElastiCache nodo e le relative informazioni sullo stato utilizzando il AWS CLI, usa il `describe-cache-cluster` comando. Ad esempio, il AWS CLI comando seguente visualizza ogni ElastiCache nodo.

```
aws elasticache describe-cache-clusters
```

Visualizzazione dello stato del ElastiCache nodo tramite l'API

Per visualizzare lo stato del ElastiCache nodo utilizzando l' ElastiCache API Amazon, chiamalo `DescribeCacheClusteroperation` con il `ShowCacheNodeInfo` flag per recuperare informazioni sui singoli nodi di cache.

Connessione ai nodi

Prima di provare a connettersi al cluster Memcached, sono necessari gli endpoint per i nodi. Per individuare gli endpoint, consultare la seguente documentazione:

- [Individuazione degli endpoint di un cluster \(console\)](#)
- [Individuazione degli endpoint \(AWS CLI\)](#)
- [Ricerca degli endpoint \(API ElastiCache\)](#)

L'esempio seguente utilizza l'utilità telnet per eseguire la connessione a un nodo che esegue Memcached.

Note

Per ulteriori informazioni su Memcached e i comandi Memcached disponibili, consulta il sito Web [Memcached](#).

Per connettersi a un nodo mediante telnet

1. Connettersi all'istanza Amazon EC2 utilizzando l'utilità di connessione preferita.

Note

Per istruzioni su come connetterti a un'istanza Amazon EC2, consulta la [Guida alle operazioni di Amazon EC2](#).

2. Scaricare e installare l'utilità telnet sull'istanza Amazon EC2. Dal prompt dei comandi dell'istanza Amazon EC2, digitare il seguente comando e specificare y al prompt dei comandi.

```
sudo yum install telnet
```

Viene visualizzato un output simile al seguente.

```
Loaded plugins: priorities, security, update-motd, upgrade-helper
Setting up Install Process
```

```
Resolving Dependencies
--> Running transaction check

...(output omitted)...

Total download size: 63 k
Installed size: 109 k
Is this ok [y/N]: y
Downloading Packages:
telnet-0.17-47.7.amzn1.x86_64.rpm                | 63 kB    00:00

...(output omitted)...

Complete!
```

3. Dal prompt dei comandi dell'istanza Amazon EC2, digitare il seguente comando, sostituendo l'endpoint del nodo con quanto mostrato in questo esempio.

```
telnet mycachecluster.eaogs8.0001.usw2.cache.amazonaws.com 11211
```

Viene visualizzato un output simile al seguente.

```
Trying 128.0.0.1...
Connected to mycachecluster.eaogs8.0001.usw2.cache.amazonaws.com.
Escape character is '^]'.
>
```

4. Eseguire il test della connessione eseguendo i comandi Memcached.

Una volta connessi a un nodo è possibile eseguire i comandi Memcached. Di seguito è riportato un esempio.

```
set a 0 0 5      // Set key "a" with no expiration and 5 byte value
hello           // Set value as "hello"
STORED
get a           // Get value for key "a"
VALUE a 0 5
hello
END
get b           // Get value for key "b" results in miss
END
>
```


Tipi di nodi supportati

Per informazioni sulle dimensioni del nodo da utilizzare, consultare [Scelta delle dimensioni dei nodi Memcached](#).

ElastiCache supporta i seguenti tipi di nodi. In generale, le generazioni attuali offrono più memoria e più potenza di elaborazione a un costo inferiore rispetto ai prodotti omologhi della generazione precedente.

Per ulteriori informazioni sui dettagli sulle prestazioni per ogni tipo di nodo, consulta [Tipi di istanza di Amazon EC2](#).

- Uso generale:
 - Generazione attuale:

Tipi di nodi M6g (disponibile solo per motore Memcached versione 1.5.16 in poi).

cache.m6g.large, cache.m6g.xlarge, cache.m6g.2xlarge, cache.m6g.4xlarge, cache.m6g.8xlarge, cache.m6g.12xlarge, cache.m6g.16xlarge

Note

Per informazioni sulla disponibilità della regione, consulta [Tipi di nodi supportati in base alla regione AWS](#).

Tipi di nodi M5: cache.m5.large, cache.m5.xlarge, cache.m5.2xlarge, cache.m5.4xlarge, cache.m5.12xlarge, cache.m5.24xlarge

Tipi di nodi M4: cache.m4.large, cache.m4.xlarge, cache.m4.2xlarge, cache.m4.4xlarge, cache.m4.10xlarge

Tipi di nodi T4g (disponibile solo per motore Memcached versione 1.5.16 e successive).

cache.t4g.micro, cache.t4g.small, cache.t4g.medium

Tipi di nodo T3: cache.t3.micro, cache.t3.small, cache.t3.medium

Tipi di nodi T2: cache.t2.micro, cache.t2.small, cache.t2.medium

- Generazione precedente: (sconsigliata. I cluster esistenti sono ancora supportati, ma la creazione di nuovi cluster non è supportata per questi tipi.)

Tipi di nodi T1: `cache.t1.micro`

Tipi di nodi M1: `cache.m1.small`, `cache.m1.medium`, `cache.m1.large`,
`cache.m1.xlarge`

Tipi di nodi M3: `cache.m3.medium`, `cache.m3.large`, `cache.m3.xlarge`,
`cache.m3.2xlarge`

- Calcolo ottimizzato:
 - Generazione precedente (non consigliata):

Tipi di nodi C1: `cache.c1.xlarge`

- Memoria ottimizzata:
 - Generazione attuale:

(i tipi di nodi R6g sono disponibili solo per motore Memcached versione 1.5.16 e successive).

Tipi di nodi R6g: `cache.r6g.large`, `cache.r6g.xlarge`, `cache.r6g.2xlarge`,
`cache.r6g.4xlarge`, `cache.r6g.8xlarge`, `cache.r6g.12xlarge`,
`cache.r6g.16xlarge`

Note

Per informazioni sulla disponibilità della regione, consulta [Tipi di nodi supportati in base alla regione AWS](#).

Tipi di nodi R5: `cache.r5.large`, `cache.r5.xlarge`, `cache.r5.2xlarge`,
`cache.r5.4xlarge`, `cache.r5.12xlarge`, `cache.r5.24xlarge`

Tipi di nodi R4: `cache.r4.large`, `cache.r4.xlarge`, `cache.r4.2xlarge`,
`cache.r4.4xlarge`, `cache.r4.8xlarge`, `cache.r4.16xlarge`

- Generazione precedente (non consigliata):

Tipi di nodi M2: `cache.m2.xlarge`, `cache.m2.2xlarge`, `cache.m2.4xlarge`

Tipi di nodi R3: `cache.r3.large`, `cache.r3.xlarge`, `cache.r3.2xlarge`,
`cache.r3.4xlarge`, `cache.r3.8xlarge`

- Ottimizzato per la rete:
- Generazione attuale:

(I tipi di nodi C7gn sono disponibili solo per motore Memcached versione 1.6.6 e successive).

Tipi di nodo C7gn: `cache.c7gn.large`, `cache.c7gn.xlarge`, `cache.c7gn.2xlarge`,
`cache.c7gn.4xlarge`, `cache.c7gn.8xlarge`, `cache.c7gn.12xlarge`,
`cache.c7gn.16xlarge`

Generazione attuale

Le tabelle seguenti mostrano la larghezza di banda di base e burst per i tipi di istanza che utilizzano il meccanismo di credito I/O di rete per superare la larghezza di banda di base.

Note

I tipi di istanza con prestazioni delle reti espandibili utilizzano un meccanismo di credito I/O di rete per superare la larghezza di banda di base sulla base del massimo sforzo.

Generale

Tipo di istanza	Versione minima supportata di Memcached	Larghezza di banda di base (Gb/s)	Larghezza di banda burst (Gb/s)
<code>cache.m7g.large</code>		0,937	12,5
<code>cache.m7g.xlarge</code>		1,876	12,5
<code>cache.m7g.2xlarge</code>		3,75	15
<code>cache.m7g.4xlarge</code>		7,5	15
<code>cache.m7g.8xlarge</code>		15	N/D

Tipo di istanza	Versione minima supportata di Memcached	Larghezza di banda di base (Gb/s)	Larghezza di banda burst (Gb/s)
cache.m7g.12xlarge		22,5	N/D
cache.m7g.16xlarge		30	N/D
cache.m6g.large	1.5.16	0.75	10,0
cache.m6g.xlarge	1.5.16	1,25	10,0
cache.m6g.2xlarge	1.5.16	2.5	10,0
cache.m6g.4xlarge	1.5.16	5.0	10,0
cache.m6g.8xlarge	1.5.16	12	N/D
cache.m6g.12xlarge	1.5.16	20	N/D
cache.m6g.16xlarge	1.5.16	25	N/D
cache.m5.large	1.5.16	0.75	10,0
cache.m5.xlarge	1.5.16	1,25	10,0
cache.m5.2xlarge	1.5.16	2.5	10,0
cache.m5.4xlarge	1.5.16	5.0	10,0
cache.m5.12xlarge	1.5.16	N/D	N/D
cache.m5.24xlarge	1.5.16	N/D	N/D
cache.m4.large	1.5.16	0,45	1.2
cache.m4.xlarge	1.5.16	0.75	2,8
cache.m4.2xlarge	1.5.16	1.0	10,0
cache.m4.4xlarge	1.5.16	2.0	10,0

Tipo di istanza	Versione minima supportata di Memcached	Larghezza di banda di base (Gb/s)	Larghezza di banda burst (Gb/s)
cache.m4.10xlarge	1.5.16	5.0	10,0
cache.t4g.micro	1.5.16	0,064	5.0
cache.t4g.small	1.5.16	0,128	5.0
cache.t4g.medium	1.5.16	0,256	5.0
cache.t3.micro	1.5.16	0,064	5.0
cache.t3.small	1.5.16	0,128	5.0
cache.t3.medium	1.5.16	0,256	5.0
cache.t2.micro	1.5.16	0,064	1,024
cache.t2.small	1.5.16	0,128	1,024
cache.t2.medium	1.5.16	0,256	1,024

Memoria ottimizzata

Tipo di istanza	Versione minima supportata	Larghezza di banda di base (Gb/s)	Larghezza di banda burst (Gb/s)
cache.r7g.large		0,937	12,5
cache.r7g.xlarge		1,876	12,5
cache.r7g.2xlarge		3,75	15
cache.r7g.4xlarge		7,5	15
cache.r7g.8xlarge		15	N/D
cache.r7g.12xlarge		22,5	N/D

Tipo di istanza	Versione minima supportata	Larghezza di banda di base (Gb/s)	Larghezza di banda burst (Gb/s)
cache.r7g.16xlarge		30	N/D
cache.r6g.large	1.5.16	0.75	10,0
cache.r6g.xlarge	1.5.16	1,25	10,0
cache.r6g.2xlarge	1.5.16	2.5	10,0
cache.r6g.4xlarge	1.5.16	5.0	10,0
cache.r6g.8xlarge	1.5.16	12	N/D
cache.r6g.12xlarge	1.5.16	20	N/D
cache.r6g.16xlarge	1.5.16	25	N/D
cache.r5.large	1.5.16	0.75	10,0
cache.r5.xlarge	1.5.16	1,25	10,0
cache.r5.2xlarge	1.5.16	2.5	10,0
cache.r5.4xlarge	1.5.16	5.0	10,0
cache.r5.12xlarge	1.5.16	20	N/D
cache.r5.24xlarge	1.5.16	25	N/D
cache.r4.large	1.5.16	0.75	10,0
cache.r4.xlarge	1.5.16	1,25	10,0
cache.r4.2xlarge	1.5.16	2.5	10,0
cache.r4.4xlarge	1.5.16	5.0	10,0
cache.r4.8xlarge	1.5.16	12	N/D
cache.r4.16xlarge	1.5.16	25	N/D

Ottimizzato per la rete

Tipo di istanza	Versione minima supportata	Larghezza di banda di base (Gb/s)	Larghezza di banda burst (Gb/s)
cache.c7gn.large	1.6.6	6,25	30
cache.c7gn.xlarge	1.6.6	12,5	40
cache.c7gn.2xlarge	1.6.6	25	50
cache.c7gn.4xlarge	1.6.6	50	N/D
cache.c7gn.8xlarge	1.6.6	100	N/D
cache.c7gn.12xlarge	1.6.6	150	N/D
cache.c7gn.16xlarge	1.6.6	200	N/D

Tipi di nodi supportati in base alla regione AWS

I tipi di nodi supportati possono variare da una regione AWS all'altra. Per ulteriori dettagli, consultare [Prezzi di Amazon ElastiCache](#).

Istanze a prestazioni espandibili

Puoi ora avviare i nodi di cache T4g, T3-Standard e T2-Standard espandibili e di scopo generico su Amazon ElastiCache. Questi nodi offrono un livello base di prestazioni della CPU con la possibilità di incrementarne l'utilizzo quando necessario fino all'esaurimento dei crediti accumulati. Un credito CPU fornisce le prestazioni di un core CPU completo per un minuto.

I nodi T4g, T3 e T2 di Amazon ElastiCache sono configurati come standard e adatti a carichi di lavoro con un utilizzo medio della CPU che è costantemente al di sotto delle prestazioni di base dell'istanza. Per superare la baseline, il nodo spende i crediti accumulati nel suo saldo del credito CPU. Se il nodo in esecuzione genera un numero basso di crediti, le prestazioni vengono gradualmente ridotte al livello base. Questa riduzione graduale assicura che il nodo non subirà un drastico calo delle prestazioni se il credito della CPU accumulato viene esaurito. Per ulteriori informazioni, consulta [Crediti CPU e prestazioni di base per istanze a prestazioni espandibili](#) nella Guida per l'utente Amazon EC2.

La tabella seguente elenca i tipi di nodi dalle prestazioni espandibili e la velocità oraria con cui vengono ottenuti i crediti CPU. Mostra anche il numero massimo di crediti CPU che un nodo può ottenere e il numero di vCPU per nodo. Inoltre, indica il livello di prestazioni di base come percentuale di prestazioni full core (utilizzando una vCPU singola).

Crediti CPU guadagnati all'ora	Quantità massima di crediti guadagnati che può essere accumulati a*	vCPU	Prestazioni di base per vCPU	Memoria (GiB)	Prestazioni di rete
12	288	2	10%	0,5	Fino a 5 Gigabit
24	576	2	20%	1,37	Fino a 5 Gigabit
24	576	2	20%	3,09	Fino a 5 Gigabit
12	288	2	10%	0,5	Fino a 5 Gigabit
24	576	2	20%	1,37	Fino a 5 Gigabit
24	576	2	20%	3,09	Fino a 5 Gigabit
6	144	1	10%	0,5	Basse a moderate
12	288	1	20%	1,55	Basse a moderate

Crediti CPU guadagnati all'ora	Quantità massima di crediti guadagnati che può essere accumulati a*	vCPU	Prestazioni di base per vCPU	Memoria (GiB)	Prestazioni di rete
24	576	2	20%	3,22	Basse a moderata

* Il numero di crediti che possono essere accumulati è equivalente al numero di crediti che possono essere guadagnati in un periodo di 24 ore.

** Le prestazioni di base nella tabella si riferiscono alla singola vCPU. Alcune dimensioni di nodo che hanno più di una vCPU. Per queste, calcolare l'utilizzo della CPU di base per il nodo, moltiplicando la percentuale di vCPU per il numero di vCPU.

Le seguenti metriche di credito CPU sono disponibili per le istanze a prestazioni espandibili T3 e T4:

Note

Queste metriche non sono disponibili per le istanze di prestazioni Burstable T2.

- CPUCreditUsage
- CPUCreditBalance

Per ulteriori informazioni su tali parametri, consulta [Parametri di credito della CPU](#).

Inoltre, ricordare i seguenti dettagli:

- Tutti i tipi di nodo della generazione corrente vengono creati in un cloud privato virtuale (VPC) basato per impostazione predefinita su Amazon VPC.

Informazioni correlate

- [Dettagli e caratteristiche del prodotto Amazon ElastiCache](#)
- [Parametri specifici del tipo di nodo Memcached](#)

Sostituzione dei nodi

Amazon ElastiCache for MemCached aggiorna frequentemente il parco istanze con patch e aggiornamenti che vengono applicati a istanze. Tuttavia, di tanto in tanto siamo costretti a lanciare nuovi nodi ElastiCache for MemCached applicare aggiornamenti obbligatori del sistema operativo dell'host. Queste sostituzioni sono necessarie per l'applicazione di aggiornamenti intesi a rafforzare sicurezza, affidabilità e prestazioni.

Offriamo l'opzione per gestire personalmente la sostituzione dei nodi nel momento che si ritiene più adatto, prima della finestra di sostituzione nodo pianificata. Quando la gestione della sostituzione è manuale, l'istanza riceve l'aggiornamento del sistema operativo quando il nodo viene riavviato e la finestra di sostituzione del nodo programmata viene annullata. Potresti continuare a ricevere avvisi che indicano l'esecuzione dell'attività di sostituzione del nodo. Se hai già ridotto manualmente la necessità di manutenzione, puoi ignorare questi avvisi.

Note

I nodi cache sostitutivi generati automaticamente da Amazon ElastiCache potrebbero avere indirizzi IP diversi. L'utente è responsabile della verifica della configurazione dell'applicazione per assicurarsi che i nodi della cache siano associati agli indirizzi IP appropriati.

Nell'elenco seguente vengono identificate le operazioni che puoi eseguire quando ElastiCache pianifica la sostituzione di uno dei nodi Memcached.

- **Non fare nulla** : Se non fai nulla, ElastiCache sostituirà il nodo come pianificato. Quando ElastiCache sostituisce automaticamente il nodo con un nuovo nodo, questo sarà inizialmente vuoto.
- **Modifica la finestra di manutenzione** : Per gli eventi di manutenzione programmata, ricevi un'e-mail o un evento di notifica da ElastiCache. In questo caso, se si modifica la finestra di manutenzione prima dell'orario di sostituzione pianificato, il nodo viene sostituito al nuovo orario. Per ulteriori informazioni, consulta [Modifica di un cluster ElastiCache](#) .

Note

La possibilità di modificare la finestra di sostituzione spostando la finestra di manutenzione è disponibile solo quando la notifica ElastiCache include una finestra di manutenzione.

Se la notifica non include una finestra di manutenzione, non puoi modificare la finestra di sostituzione.

Supponiamo, ad esempio, che sia giovedì 9 novembre alle 15:00 e che la prossima finestra di manutenzione sia prevista per venerdì 10 novembre alle 17:00. Questi sono tre possibili scenari e i relativi risultati:

- Sposti la finestra di manutenzione alle ore 16:00 di venerdì, dopo la data e l'ora corrente e prima della prossima finestra di manutenzione programmata. Il nodo viene sostituito venerdì 10 novembre alle 16:00.
- Sposti la finestra di manutenzione alle ore 16:00 di sabato, dopo la data e l'ora corrente e dopo la prossima finestra di manutenzione programmata. Il nodo viene sostituito sabato 11 novembre alle 16:00.
- Sposti la finestra di manutenzione alle ore 16:00 di mercoledì, anticipandola rispetto alla data e ora corrente. Il nodo viene sostituito il prossimo mercoledì 15 novembre alle 16:00.

Per istruzioni, consultare [Gestione della manutenzione](#).

- Sostituire manualmente il nodo : Se è necessario sostituire il nodo prima della finestra di manutenzione successiva, sostituire manualmente il nodo.

Se sostituisci manualmente il nodo, le chiavi vengono ridistribuite. La ridistribuzione causa degli eventi di cache miss.

Per sostituire manualmente un nodo Memcached

1. Eliminare il nodo pianificato per la sostituzione. Per istruzioni, consultare [Rimozione di nodi da un cluster](#).
2. Aggiungere un nuovo nodo al cluster. Per istruzioni, consultare [Aggiunta di nodi a un cluster](#).
3. Se non stai utilizzando l'individuazione automatica sul cluster, vedi l'applicazione e sostituisci ogni istanza dell'endpoint del nodo precedente con l'endpoint del nuovo nodo.

Nodi riservati ElastiCache

La prenotazione di uno o più nodi può essere un modo per ridurre i costi. I nodi riservati vengono addebitati un costo anticipato che dipende dal tipo di nodo e dalla durata della prenotazione, uno o tre anni.

Per verificare se i nodi riservati comportano un risparmio sui costi per i propri casi d'uso, determinare innanzitutto la dimensione del nodo e il numero di nodi necessari. Quindi stimare l'utilizzo del nodo e confrontare il costo totale risultante dall'utilizzo dei nodi on demand rispetto all'utilizzo dei nodi riservati. Puoi combinare l'utilizzo di nodi prenotati e on demand nei cluster. Per informazioni sui prezzi, consulta [Prezzi Amazon ElastiCache](#).

Note

I nodi riservati non sono flessibili; si applicano solo al tipo di istanza esatto che si riserva.

Gestione dei costi con nodi riservati

Prenotare uno o più nodi può essere un modo per ridurre di costi. I nodi riservati vengono addebitati un costo anticipato che dipende dal tipo di nodo e dalla durata della prenotazione, ovvero uno o tre anni. Tale addebito è molto inferiore all'addebito orario di utilizzo previsto in caso di nodi on demand.

Per verificare se i nodi riservati comportano un risparmio sui costi per i propri casi d'uso, determinare innanzitutto la dimensione del nodo e il numero di nodi necessari. Quindi stimare l'utilizzo del nodo e confrontare il costo totale risultante dall'utilizzo dei nodi on demand rispetto all'utilizzo dei nodi riservati. Puoi combinare l'utilizzo di nodi prenotati e on demand nei cluster. Per informazioni sui prezzi, consulta [Prezzi Amazon ElastiCache](#).

Regione AWS, tipo di nodo e durata devono essere scelti al momento dell'acquisto e non possono essere modificati in un secondo momento.

Puoi utilizzare AWS Management Console, AWS CLI o l'API ElastiCache per elencare e acquistare le offerte di nodi riservati disponibili.

Per ulteriori informazioni sui nodi riservati, consulta [Nodi riservati di Amazon ElastiCache](#).

Argomenti

- [Offerte di nodi riservati standard](#)
- [Offerte di nodi riservati legacy](#)

- [Recupero di informazioni sulle offerte di nodi riservati](#)
- [Acquisto di un nodo riservato](#)
- [Recupero di informazioni sui nodi riservati](#)

Offerte di nodi riservati standard

Quando acquisti un'istanza di nodi standard riservata (RI) in Amazon ElastiCache, acquisti un impegno che ti permette di ottenere una tariffa scontata per un tipo di istanza di nodo di cache e una regione AWS specifici, per la durata dell'istanza di nodo riservata. Per usare un'istanza database di nodo riservata Amazon ElastiCache, devi creare un'istanza di nodo ElastiCache con una procedura analoga a quella per la creazione di un'istanza on demand.

La nuova istanza di nodo creata deve corrispondere esattamente alle specifiche dell'istanza di nodo riservata. Se le specifiche della nuova istanza di nodo corrispondono a un'istanza di nodo riservata esistente per il tuo account, viene fatturata la tariffa scontata per l'istanza di nodo riservata. In caso contrario, l'istanza di nodo viene fatturata in base a una tariffa on demand. Le RI standard disponibili a partire dalle famiglie di istanze R5 and M5 in poi.

Note

Tutti e tre i tipi di offerta illustrati qui di seguito sono disponibili per un periodo di un anno o di tre anni.

Tipi offerta

Nessun pagamento anticipato RI permette di accedere a un'istanza ElastiCache riservata senza un pagamento anticipato. L'istanza riservata ElastiCache senza pagamento anticipato viene fatturata applicando una tariffa oraria scontata per ogni ora entro il termine, indipendentemente dall'utilizzo.

Pagamento anticipato parziale RI richiede il pagamento anticipato di una parte dell'istanza ElasticCache riservata. Le ore rimanenti del periodo di prenotazione vengono fatturate a una tariffa oraria scontata, indipendentemente dall'utilizzo. Questa opzione sostituisce l'opzione precedente **Utilizzo elevato**, illustrata nella sezione seguente.

Pagamento anticipato intero costo il servizio RI richiede il pagamento completo all'inizio del mandato del RI. Il pagamento viene effettuato per intero all'inizio del periodo della prenotazione e non vengono addebitati altri costi per il resto del periodo, indipendentemente dal numero di ore di utilizzo.

Offerte di nodi riservati legacy

Esistono tre livelli di prenotazioni dei nodi legacy: utilizzo pesante, utilizzo medio e utilizzo leggero. I nodi possono essere riservati a qualsiasi livello di utilizzo per 1 o 3 anni. Tipo di nodo, livello di utilizzo e periodo di prenotazione influiscono sui costi totali. Prima di acquistare i nodi riservati, confronta i vari modelli per verificare il risparmio che ciascun modello può garantire alla tua attività.

I nodi acquistati in base a un livello o periodo di utilizzo non possono essere convertiti in un altro livello o periodo di utilizzo.

Livelli di utilizzo

Nodi riservati con utilizzo Heavy consentono carichi di lavoro con una base di capacità coerente o eseguono carichi di lavoro con stato costante. I nodi riservati con utilizzo Heavy richiedono un elevato costo anticipato, ma se prevedi un'esecuzione per più del 79 percento del periodo contrattualizzato di nodi riservati puoi ottenere un maggiore risparmio (fino al 70 percento in meno sul prezzo on demand). I nodi riservati con utilizzo pesante prevedono un singolo addebito. Questo viene quindi seguito da una tariffa oraria più bassa per la durata del periodo, indipendentemente dal fatto che il nodo sia in esecuzione.

I nodi riservati con utilizzo Medium rappresentano l'opzione migliore se prevedi di utilizzare i nodi riservati per una notevole quantità di tempo, ma desideri un singolo addebito di minore entità o la flessibilità di interrompere il pagamento del nodo una volta spento. I nodi riservati con utilizzo Medium rappresentano l'opzione più conveniente se prevedi un'esecuzione per più del 40 percento del periodo contrattualizzato di nodi riservati. Questa opzione ti consente di risparmiare fino al 64 percento sul prezzo on demand. Con i nodi riservati con utilizzo Medium, paghi un singolo addebito leggermente superiore rispetto ai nodi riservati con utilizzo Light e disponi di tariffe orarie di utilizzo inferiori applicabili durante l'esecuzione di un nodo.

I nodi riservati con utilizzo Light sono ideali per carichi di lavoro periodici eseguiti solo un paio di ore al giorno o alcuni giorni a settimana. I nodi riservati con utilizzo Light prevedono un singolo addebito e una tariffa oraria di utilizzo scontata applicabile quando il nodo è in esecuzione. Il risparmio inizia a essere evidente quando il nodo viene eseguito per più del 17 percento del periodo contrattuale di nodi riservati. Per l'intero periodo del nodo riservato, puoi risparmiare fino al 56 percento delle tariffe on demand.

Offerte di nodi riservati legacy

Offerta	Costo anticipato	Tariffa di utilizzo	Vantaggio
---------	------------------	---------------------	-----------

Offerta	Costo anticipato	Tariffa di utilizzo	Vantaggio
Utilizzo pesante	Massimo	Tariffa oraria minore. Applicabile all'intero periodo indipendentemente dall'utilizzo del nodo riservato.	Costo complessivo inferiore se prevedi di eseguire i nodi riservati per oltre il 79 per cento di un periodo di tre anni.
Medium utilizzo	Media	Tariffa oraria di utilizzo applicata per ogni ora di esecuzione e del nodo. Nessun costo orario applicato se il nodo non viene eseguito.	Ideale per carichi di lavoro elastici o quando prevedi un utilizzo moderato, per oltre il 40 per cento di un periodo di tre anni.
Utilizzo Light	Minimo	Tariffa oraria di utilizzo applicata per ogni ora di esecuzione e del nodo. Nessun costo orario applicato se il nodo non viene eseguito. Tariffe orarie più alte rispetto a tutti i tipi di offerta, sebbene vengano applicate solo se il nodo riservato è in esecuzione.	Costo complessivo più alto se si prevede di eseguire tutto il tempo. Costo complessivo più elevato se prevedi un'esecuzione continuativa. Il costo complessivo sarà invece minimo se prevedi un utilizzo poco frequente del nodo riservato, superiore al 15 per cento di un periodo di tre anni.

Offerta	Costo anticipato	Tariffa di utilizzo	Vantaggio
Uso su richiesta (nessun nodo riservato)	Nessuno	Tariffa oraria maggiore. Applicata ogni volta che il nodo è in esecuzione.	Costo orario maggiore.

Per ulteriori informazioni, consulta [Prezzi di Amazon ElastiCache](#).

Recupero di informazioni sulle offerte di nodi riservati

Puoi ottenere informazioni sulle offerte disponibili per i nodi riservati prima di effettuarne l'acquisto.

I seguenti esempi mostrano come ottenere i prezzi e i dettagli delle offerte di nodi riservati disponibili tramite la AWS Management Console, la AWS CLI e l'API ElastiCache.

Argomenti

- [Recupero di informazioni sulle offerte di nodi riservati](#)
- [Recupero di informazioni sulle offerte di nodi riservati \(AWS CLI\)](#)
- [Recupero di informazioni sulle offerte di nodi riservati \(API ElastiCache\)](#)

Recupero di informazioni sulle offerte di nodi riservati

Per ottenere i prezzi e altre informazioni sulle offerte di cluster riservati disponibili tramite la AWS Management Console, usa la procedura seguente.

Per recuperare informazioni sulle offerte di nodi riservati disponibili

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nel pannello di navigazione scegliere Nodi riservati.
3. Scegliere Purchase Reserved Nodes (Acquista nodi riservati).
4. In Engine (Motore), scegliere Memcached.
5. Per determinare le offerte disponibili, effettua selezioni per le opzioni seguenti:
 - Tipo di nodo
 - Termine
 - Tipo offerta

Dopo avere effettuato le selezioni desiderate, il costo per nodo e il costo totale delle tue selezioni viene mostrato in Dettagli di prenotazione.

6. Scegliere Cancel (Annulla) per evitare di acquistare tali nodi e incorrere nel conseguente addebito dei relativi costi.

Recupero di informazioni sulle offerte di nodi riservati (AWS CLI)

Per ottenere i prezzi e altre informazioni sulle offerte di nodi riservati disponibili, digita il comando seguente al prompt dei comandi.

```
aws elasticache describe-reserved-cache-nodes-offerings
```

Questa operazione genera un output simile al seguente (formato JSON):

```
{
  "ReservedCacheNodesOfferingId": "0xxxxxxxx-xxeb-44ex-xx3c-xxxxxxxx072",
  "CacheNodeType": "cache.xxx.large",
  "Duration": 94608000,
  "FixedPrice": XXXX.X,
  "UsagePrice": X.X,
  "ProductDescription": "memcached",
  "OfferingType": "All Upfront",
  "RecurringCharges": [
    {
      "RecurringChargeAmount": X.X,
      "RecurringChargeFrequency": "Hourly"
    }
  ]
},
{
  "ReservedCacheNodesOfferingId": "0xxxxxxxx-xxeb-44ex-xx3c-xxxxxxxx072",
  "CacheNodeType": "cache.xxx.xlarge",
  "Duration": 94608000,
  "FixedPrice": XXXX.X,
  "UsagePrice": X.X,
  "ProductDescription": "memcached",
  "OfferingType": "Partial Upfront",
  "RecurringCharges": [
    {
      "RecurringChargeAmount": X.XXXX,
      "RecurringChargeFrequency": "Hourly"
    }
  ]
},
{
  "ReservedCacheNodesOfferingId": "0xxxxxxxx-xxeb-44ex-xx3c-xxxxxxxx072",
  "CacheNodeType": "cache.xx.12xlarge",
  "Duration": 31536000,
```



```
    "FixedPrice": X.X,  
    "UsagePrice": X.X,  
    "ProductDescription": "memcached",  
    "OfferingType": "No Upfront",  
    "RecurringCharges": [  
      {  
        "RecurringChargeAmount": X.XXXX,  
        "RecurringChargeFrequency": "Hourly"  
      }  
    ]  
  }  
}
```

Per ulteriori informazioni, consulta [describe-reserved-cache-nodes-offerings](#) nella documentazione di riferimento di AWS CLI.

Recupero di informazioni sulle offerte di nodi riservati (API ElastiCache)

Per ottenere i prezzi e altre informazioni sulle offerte di nodi riservati disponibili, chiama l'operazione `DescribeReservedCacheNodesOfferings`.

Example

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DescribeReservedCacheNodesOfferings  
&Version=2014-12-01  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20141201T220302Z  
&X-Amz-Algorithm  
&X-Amz-SignedHeaders=Host  
&X-Amz-Expires=20141201T220302Z  
&X-Amz-Credential=<credential>  
&X-Amz-Signature=<signature>
```

Per ulteriori informazioni, consulta [DescribeReservedCacheNodesOfferings](#) nella Guida di riferimento delle API ElastiCache.

Acquisto di un nodo riservato

Gli esempi seguenti mostrano come acquistare un'offerta di nodo riservato tramite la AWS Management Console, AWS CLI e l'API ElastiCache.

Important

Le procedure negli esempi seguenti implicano l'addebito sull'account AWS di costi che non potranno essere annullati.

Argomenti

- [Acquisto di un nodo riservato \(Console\)](#)
- [Acquisto di un nodo riservato \(AWS CLI\)](#)
- [Acquisto di un nodo riservato \(API ElastiCache\)](#)

Acquisto di un nodo riservato (Console)

Questo esempio mostra come acquistare un'offerta di nodi riservati specifica, 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f, con ID nodo riservato myreservationID.

La seguente procedura prevede l'utilizzo della AWS Management Console per acquistare l'offerta di nodi riservati per ID offerta.

Per acquistare nodi riservati

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nel pannello di navigazione scegliere il collegamento Reserved Nodes (Nodi riservati).
3. Scegliere il pulsante Purchase Reserved Nodes (Acquista nodi riservati).
4. In Engine (Motore), scegliere Memcached.
5. Per determinare le offerte disponibili, effettua selezioni per le opzioni seguenti:
 - Tipo di nodo
 - Termine
 - Tipo offerta
 - Un ID nodo riservato opzionale

Dopo avere effettuato le selezioni desiderate, il costo per nodo e il costo totale delle tue selezioni vengono mostrati in Dettagli di prenotazione.

6. Scegliere Purchase (Acquista).

Acquisto di un nodo riservato (AWS CLI)

L'esempio seguente mostra come acquistare l'offerta di cluster riservati specifica, 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f, con ID nodo riservato myreservationID.

Al prompt dei comandi, digita il comando seguente.

Per Linux, macOS o Unix:

```
aws elasticache purchase-reserved-cache-nodes-offering \  
  --reserved-cache-nodes-offering-id 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f \  
  --reserved-cache-node-id myreservationID
```

Per Windows:

```
aws elasticache purchase-reserved-cache-nodes-offering ^  
  --reserved-cache-nodes-offering-id 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f ^  
  --reserved-cache-node-id myreservationID
```

Il comando restituisce un output simile al seguente:

RESERVATION	ReservationId	Class	Start Time	Duration	
Fixed Price	Usage Price	Count	State	Description	Offering Type
RESERVATION	myreservationid	cache.xx.small	2013-12-19T00:30:23.247Z	1y	
XXX.XX USD	X.XXX USD	1	payment-pending	memcached	Medium Utilization

Per ulteriori informazioni, consulta [purchase-reserved-cache-nodes-offering](#) nella documentazione di riferimento ad AWS CLI.

Acquisto di un nodo riservato (API ElastiCache)

L'esempio seguente mostra come acquistare l'offerta di nodi riservati specifica, 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f, con ID cluster riservato myreservationID.

Chiama l'operazione PurchaseReservedCacheNodesOffering con i parametri seguenti:

- ReservedCacheNodesOfferingId = 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f
- ReservedCacheNodeID = myreservationID
- CacheNodeCount = 1

Example

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=PurchaseReservedCacheNodesOffering  
  &ReservedCacheNodesOfferingId=649fd0c8-cf6d-47a0-bfa6-060f8e75e95f  
  &ReservedCacheNodeID=myreservationID  
  &CacheNodeCount=1  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20141201T220302Z  
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
  &X-Amz-Date=20141201T220302Z  
  &X-Amz-SignedHeaders=Host  
  &X-Amz-Expires=20141201T220302Z  
  &X-Amz-Credential=<credential>  
  &X-Amz-Signature=<signature>
```

Per ulteriori informazioni, consulta [PurchaseReservedCacheNodesOffering](#) nella Guida di riferimento delle API ElastiCache.

Recupero di informazioni sui nodi riservati

Puoi ottenere informazioni sui nodi riservati che hai acquistato tramite la AWS Management Console, la AWS CLI e l'API ElastiCache.

Argomenti

- [Recupero di informazioni sui nodi riservati \(Console\)](#)
- [Recupero di informazioni sui nodi riservati \(AWS CLI\)](#)
- [Recupero di informazioni sui nodi prenotati \(API Elasticache\)](#)

Recupero di informazioni sui nodi riservati (Console)

La procedura seguente illustra come utilizzare la AWS Management Console per ottenere informazioni sui nodi riservati che hai acquistato.

Per ottenere informazioni sui nodi riservati che hai acquistato

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nel pannello di navigazione scegliere il collegamento Reserved Nodes (Nodi riservati).

I nodi riservati del tuo account sono presenti nell'elenco Nodi riservati. Puoi scegliere qualsiasi nodo riservato nell'elenco per visualizzare le relative informazioni dettagliate nel riquadro dei dettagli nella parte inferiore della console.

Recupero di informazioni sui nodi riservati (AWS CLI)

Per ottenere informazioni sui nodi riservati per l'account AWS, digita il seguente comando al prompt dei comandi:

```
aws elasticache describe-reserved-cache-nodes
```

Questa operazione genera un output simile al seguente (formato JSON):

```
{
  "ReservedCacheNodeId": "myreservationid",
  "ReservedCacheNodesOfferingId": "649fd0c8-cf6d-47a0-bfa6-060f8e75e95f",
  "CacheNodeType": "cache.xx.small",
```

```
"Duration": "31536000",
"ProductDescription": "memcached",
"OfferingType": "Medium Utilization",
"MaxRecords": 0
}
```

Per ulteriori informazioni, consulta [describe--reserved-cache-nodes](#) nella documentazione di riferimento ad AWS CLI.

Recupero di informazioni sui nodi prenotati (API Elasticache)

Per ottenere informazioni sui nodi riservati per l'account AWS in uso, chiama l'operazione `DescribeReservedCacheNodes`.

Example

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeReservedCacheNodes
&Version=2014-12-01
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20141201T220302Z
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256
&X-Amz-Date=20141201T220302Z
&X-Amz-SignedHeaders=Host
&X-Amz-Expires=20141201T220302Z
&X-Amz-Credential=<credential>
&X-Amz-Signature=<signature>
```

Per ulteriori informazioni, consulta [DescribeReservedCacheNodes](#) nella Guida di riferimento delle API ElastiCache.

Migrazione dei nodi della generazione precedente

I nodi di generazione precedente sono tipi di nodi che vengono eliminati gradualmente. Se non si dispone di cluster esistenti che utilizzano un tipo di nodo di generazione precedente, ElastiCache non supporta la creazione di nuovi cluster con tale tipo di nodo.

A causa della quantità limitata di tipi di nodi della generazione precedente, non è possibile garantire una sostituzione corretta quando un nodo diventa non integro nei cluster. In tale scenario, la disponibilità del cluster potrebbe essere influenzata negativamente.

Si consiglia di eseguire la migrazione dei cluster in un nuovo tipo di nodo per migliorare la disponibilità e le prestazioni. Per un tipo di nodo consigliato a cui eseguire la migrazione, vedere [Percorsi di aggiornamento](#). Per un elenco completo dei tipi di nodo supportati e dei tipi di nodo della generazione precedente in ElastiCache, vedere [Tipi di nodi supportati](#).

Migrazione dei nodi in un cluster Memcached

Per eseguire la migrazione di ElastiCache for MemCached a un tipo di nodo diverso, è necessario creare un nuovo cluster, che inizia sempre vuoto che l'applicazione può popolare.

Per eseguire la migrazione del tipo di nodo del cluster ElastiCache for MemCached utilizzando ElastiCache Console:

- Creare un nuovo cluster con il nuovo tipo di nodo. Per ulteriori informazioni, consulta [Creazione di un cluster Memcached \(console\)](#).
- Nell'applicazione, aggiornare gli endpoint con gli endpoint del nuovo cluster. Per ulteriori informazioni, consulta [Individuazione degli endpoint di un cluster \(console\)](#)
- Eliminare il cluster precedente. Per ulteriori informazioni, consulta [Eliminazione di un cluster](#)

Lavorare con ElastiCache

In questa sezione puoi trovare dettagli su come gestire i vari componenti dell' ElastiCache implementazione.

Argomenti

- [Versioni e aggiornamenti del motore](#)
- [Best practice e strategie di caching ElastiCache](#)
- [Gestione del cluster progettato autonomamente](#)
- [Scalabilità ElastiCache per Memcached](#)
- [Assegnazione di tag delle risorse ElastiCache](#)
- [Utilizzo di Amazon ElastiCache Well-Architected Lens](#)
- [Procedure e best practice comuni per la risoluzione dei problemi](#)
- [Ulteriori passaggi per la risoluzione dei problemi](#)

Versioni e aggiornamenti del motore

Questa sezione illustra e spiega come aggiornare le versioni del motore Memcached supportate.

Argomenti

- [ElastiCache supportata per le versioni Memcached](#)
- [Versioni e aggiornamenti del motore](#)
- [Come aggiornare la versione di un motore](#)

ElastiCache supportata per le versioni Memcached

ElastiCache supporta le seguenti versioni Memcached e i relativi aggiornamenti. Per effettuare l'aggiornamento a una versione più recente, occorre prestare particolare attenzione alle condizioni che, se non soddisfatte, ne possono causare la mancata esecuzione.

ElastiCache per le versioni Memcached

- [Memcached versione 1.6.22](#)
- [Memcached versione 1.6.17](#)
- [Memcached versione 1.6.12](#)
- [Memcached versione 1.6.6](#)
- [Memcached versione 1.5.16](#)
- [Memcached versione 1.5.10](#)
- [Memcached versione 1.4.34](#)
- [Memcached versione 1.4.33](#)
- [Memcached versione 1.4.24](#)
- [Memcached versione 1.4.14](#)
- [Memcached versione 1.4.5](#)

Memcached versione 1.6.22

ElastiCache per Memcached aggiunge il supporto per Memcached versione 1.6.22. Non include nuove funzionalità, ma include correzioni di bug e aggiornamenti cumulativi da [Memcached 1.6.18](#).

Per ulteriori informazioni, consulta [ReleaseNotes1622](#) in Memcached su GitHub.

Memcached versione 1.6.17

ElastiCache per Memcached aggiunge il supporto per Memcached versione 1.6.17. Non include nuove funzionalità, ma include correzioni di bug e aggiornamenti cumulativi da [Memcached 1.6.17](#).

Per ulteriori informazioni, consultare [Note di rilascio 1617](#) in Memcached su GitHub.

Memcached versione 1.6.12

ElastiCache per Memcached aggiunge il supporto per Memcached versione 1.6.12 e crittografia dei dati in transito. Inoltre, include correzioni di bug e aggiornamenti cumulativi da [Memcached 1.6.6](#).

Per ulteriori informazioni, consulta le [Note di rilascio 1612](#) in Memcached su GitHub.

Memcached versione 1.6.6

ElastiCache for Memcached aggiunge il support per Memcached versione 1.6.6. Non include nuove caratteristiche, ma include correzioni di bug e aggiornamenti cumulativi da [Memcached 1.5.16](#).

ElastiCache for Memcached non include il support per [Extstore](#).

Per ulteriori informazioni, consulta le [note di rilascio di Memcached 166](#) nella sezione dedicata a Memcached su GitHub.

Memcached versione 1.5.16

ElastiCache for Memcached aggiunge support per Memcached versione 1.5.16. Non include nuove caratteristiche, ma include correzioni di bug e aggiornamenti cumulativi da [Memcached 1.5.14](#) e [Memcached 1.5.15](#).

Per ulteriori informazioni, consulta le [Note di rilascio di Memcached 1.5.16](#) nella sezione dedicata a Memcached su GitHub.

Memcached versione 1.5.10

ElastiCache for Memcached versione 1.5.10 supporta le seguenti caratteristiche Memcached:

- Ribilanciamento automatizzato dell'allocazione.
- Ricerche più rapide nella tabella hash con l'algoritmo murmur3.
- Algoritmo LRU segmentato.
- Crawler LRU alla memoria di recupero in background.
- `--enable-seccomp`: un'opzione della fase di compilazione.

Introduce anche i parametri `no_modern` e `inline_ascii_resp`. Per ulteriori informazioni, consulta [Modifiche dei parametri di Memcached 1.5.10](#).

Tra i miglioramenti di Memcached aggiunti a ElastiCache for Memcached a partire dalla versione 1.4.34 sono inclusi:

- Correzioni cumulative come ASCII multiget, CVE-2017-9951 e crawl dei limiti per `metadumper`.
- Migliore gestione delle connessioni mediante la chiusura delle connessioni una volta raggiunto il limite di connessioni.

- Gestione migliorata delle dimensioni per elementi di dimensioni maggiori di 1 MB.
- Migliori prestazioni e miglioramenti del sovraccarico di memoria mediante la riduzione di qualche byte dei requisiti di memoria per voce.

Per ulteriori informazioni, consulta le [note di rilascio di Memcached 1.5.10](#) nella sezione dedicata a Memcached su GitHub.

Memcached versione 1.4.34

ElastiCache for Memcached versione 1.4.34 non prevede nuove funzioni rispetto alla versione 1.4.33. La 1.4.34 è una versione di correzione dei bug più estesa del solito.

Per ulteriori informazioni, consulta le [note di rilascio di Memcached 1.4.34](#) nella sezione dedicata a Memcached su GitHub.

Memcached versione 1.4.33

Tra i miglioramenti di Memcached applicati a partire dalla versione 1.4.24 sono inclusi:

- Capacità di esecuzione del dump dei metadati afferenti a una classe, un elenco di classi o tutte le classi di slab. Per ulteriori informazioni, consulta le [note di rilascio di Memcached 1.4.31](#).
- Supporto migliorato per elementi di grandi dimensioni, superiori al megabyte di default. Per ulteriori informazioni, consulta le [note di rilascio di Memcached 1.4.29](#).
- Capacità di stabilire quanto tempo un client può rimanere inattivo prima che ne venga richiesta la chiusura.

Capacità di aumentare dinamicamente la quantità di memoria disponibile per Memcached senza dover riavviare il cluster. Per ulteriori informazioni, consulta le [note di rilascio di Memcached 1.4.27](#).

- Registrazione di fetchers, mutations ed evictions ora supportata. Per ulteriori informazioni, consulta le [note di rilascio di Memcached 1.4.26](#).
- Memoria liberata recuperabile in un pool globale e riassegnabile a nuove classi di slab. Per ulteriori informazioni, consulta le [note di rilascio di Memcached 1.4.25](#).
- Risoluzione di vari bug.
- Nuovi comandi e parametri. Per un elenco, consulta [Parametri di Memcached 1.4.33 aggiunti](#).

Memcached versione 1.4.24

Tra i miglioramenti di Memcached applicati a partire dalla versione 1.4.14 sono inclusi:

- Gestione della cache utilizzata meno di recente (LRU) come attività in background.
- Possibilità di utilizzo di jenkins o murmur3 come algoritmo hash.
- Nuovi comandi e parametri. Per un elenco, consulta [Parametri di Memcached 1.4.24 aggiunti](#).
- Risoluzione di vari bug.

Memcached versione 1.4.14

Tra i miglioramenti di Memcached applicati a partire dalla versione 1.4.5 sono inclusi:

- Capacità potenziata di ribilanciamento dell'allocazione.
- Miglioramenti nelle prestazioni e nella scalabilità.
- Introduzione del comando touch, utile per aggiornare la scadenza di un elemento esistente senza il recupero.
- Auto discovery - consente ai programmi client di identificare automaticamente tutti i nodi cache in un cluster e di avviare e mantenere le connessioni a tutti questi nodi.

Memcached versione 1.4.5

Memcached 1.4.5 è il primo motore, nonché versione, supportato da Amazon ElastiCache for Memcached.

Versioni e aggiornamenti del motore

Le versioni MAJOR sono per modifiche incompatibili con l'API e le versioni MINOR sono per nuove funzionalità aggiunte per compatibilità a ritroso. Le versioni PATCH sono per correzioni di bug compatibili a ritroso e modifiche non funzionali.

Gestione delle versioni per ElastiCache serverless

ElastiCache serverless applica automaticamente alla cache l'ultima versione SECONDARIA e della PATCH del software, senza alcun impatto o tempi di inattività dell'applicazione. Non è richiesta nessuna azione da parte tua.

Quando è disponibile una nuova versione PRINCIPALE, ElastiCache serverless invia una notifica nella console e un evento in EventBridge. Puoi scegliere di aggiornare la cache all'ultima versione principale utilizzando la console, la CLI o l'API e selezionando la versione più recente del motore.

Gestione delle versioni per cluster ElastiCache progettati autonomamente

Quando si usano cluster ElastiCache progettati autonomamente, è possibile controllare quando aggiornare il software di un cluster di cache a una nuova versione supportata da ElastiCache. Puoi controllare quando aggiornare la cache all'ultima versione PRINCIPALE, SECONDARIA e della PATCH disponibili. L'utente può eseguire l'aggiornamento a una versione del motore sul cluster o gruppo di replica modificando quest'ultimo e specificando la nuova versione da utilizzare.

Puoi controllare se e quando aggiornare il software conforme al protocollo di un cluster di cache a una nuova versione supportata da ElastiCache. Questo livello di controllo ti consente di mantenere la compatibilità con versioni specifiche, testare le nuove versioni con l'applicazione prima di distribuirle in produzione e aggiornare le versioni alle tue condizioni e secondo le tue scadenze.

Poiché presentano rischi relativi alla compatibilità, gli aggiornamenti delle versioni non vengono eseguiti automaticamente, ma devono essere avviati manualmente.

Per eseguire l'aggiornamento a una versione più recente di Memcached, occorre modificare il cluster di cache specificando la nuova versione del motore che si desidera utilizzare. L'aggiornamento a una versione più recente di Memcached è un processo distruttivo: si perdono i dati e si inizia con una cache a freddo. Per ulteriori informazioni, consulta [Modifica dei cluster](#).

Quando viene eseguito l'aggiornamento da una versione precedente alla versione 1.4.33 o una successiva di Memcached, è importante tenere presente i requisiti riportati di seguito. `CreateCacheCluster` e `ModifyCacheCluster` non riescono nelle condizioni seguenti:

- Se `slab_chunk_max > max_item_size`.
- Se `max_item_size` modulo `slab_chunk_max` $\neq 0$.
- Se `max_item_size > ((max_cache_memory - memcached_connections_overhead) / 4)`.

Il valore `(max_cache_memory - memcached_connections_overhead)` rappresenta la memoria del nodo utilizzabile per i dati. Per ulteriori informazioni, consulta [Sovraccarico delle connessioni Memcached](#).

Considerazioni sull'aggiornamento quando si usano cluster progettati autonomamente

Note

Le seguenti considerazioni si applicano solo quando si aggiornano cluster progettati autonomamente. Non si applicano a ElastiCache serverless.

Quando aggiorni un cluster progettato autonomamente, considera quanto indicato di seguito.

- La gestione della versione del motore è progettata in modo da avere il maggior controllo possibile sulle modalità di applicazione delle patch. Tuttavia, ElastiCache si riserva il diritto di applicare patch al cluster per tuo conto in caso di vulnerabilità critiche della sicurezza nel software del sistema e della cache.
- Poiché il motore Memcached non prevede la persistenza, l'aggiornamento a una particolare versione è sempre un processo radicale, che cancella tutti i dati della cache nel cluster.

Come aggiornare la versione di un motore

Per avviare gli aggiornamenti della versione al cluster, modificalo e specifica una versione del motore più recente. A questo scopo, utilizza la console ElastiCache, l'AWS CLI o l'API ElastiCache:

- Per utilizzare la AWS Management Console, consulta [Modifica dei cluster mediante la console](#).
- Per utilizzare la AWS CLI, consulta [Modifica dei cluster mediante la CLI](#).
- Per utilizzare l'API ElastiCache, consulta [Modifica dei cluster mediante l'API](#).

Come aggiornare la versione di un motore

Per avviare gli aggiornamenti della versione al cluster, modificalo e specifica una versione del motore più recente. A questo scopo, utilizza la console ElastiCache, l'AWS CLI o l'API ElastiCache:

- Per utilizzare la AWS Management Console, consulta – [Utilizzando il AWS Management Console](#).
- Per utilizzare l'AWS CLI, consulta [Utilizzando il AWS CLI](#).
- Per utilizzare l'API ElastiCache, vedere [Utilizzo dell' ElastiCache API](#).

Best practice e strategie di caching ElastiCache

Di seguito sono indicate le best practice consigliate per l'utilizzo di Amazon ElastiCache. Queste best practice consentono di migliorare prestazioni e affidabilità della cache.

Argomenti

- [Best practice con i client Memcached](#)
- [Comandi Memcached supportati](#)
- [Strategie di caching](#)

Best practice con i client Memcached

Per ulteriori informazioni sulle best practice per l'interazione delle risorse ElastiCache con le librerie client Memcached open source di uso comune, consulta gli argomenti seguenti.

Argomenti

- [Configurazione del client ElastiCache per un efficiente bilanciamento del carico](#)
- [Esempi di client IPv6](#)

Configurazione del client ElastiCache per un efficiente bilanciamento del carico

Note

Questa sezione si applica ai cluster Memcached a più nodi progettati autonomamente.

Per utilizzare in modo efficace più nodi ElastiCache Memcached, devi essere in grado di distribuire le chiavi di cache tra i nodi. Un modo semplice per eseguire il bilanciamento del carico di un cluster con n nodi consiste nel calcolare l'hash della chiave dell'oggetto ed eseguire il comando `mod` sul risultato per $n - \text{hash}(\text{key}) \bmod n$. Il valore risultante (da 0 a $n-1$) è il numero del nodo in cui inserisci l'oggetto.

Questo approccio è semplice ed efficace, purché il numero dei nodi (n) sia costante. Quando aggiungi o rimuovi un nodo dal cluster, il numero di chiavi da spostare corrisponde tuttavia a $(n - 1) / n$ (dove n è il nuovo numero di nodi). Con questo approccio, il numero di chiavi da spostare è pertanto maggiore, con un conseguente numero elevato di mancati riscontri nella cache iniziali, specialmente man mano che aumenta il numero dei nodi. Con il dimensionamento da 1 a 2 nodi, viene spostato un numero di chiavi pari a $(2-1)/2$ (50 percento), ovvero il caso migliore. Con il dimensionamento da 9 a 10 nodi, viene spostato un numero di chiavi pari a $(10-1)/10$ (90 percento). In caso di un dimensionamento dovuto a un picco nel traffico, è bene evitare un elevato numero di mancati riscontri nella cache, in quanto ciò causerebbe hit nel database, già sovraccaricato dai picchi di traffico.

La soluzione a tale problema è rappresentata da un hashing coerente. L'hashing coerente utilizza un algoritmo in modo tale che quando un nodo viene aggiunto o rimosso da un cluster, il numero di chiavi da spostare è circa $1/n$ (dove n è il nuovo numero di nodi). Passando da 1 a 2 nodi, viene spostato un numero di chiavi pari a $1/2$ (50 percento), ovvero il caso peggiore. Passando da 9 a 10 nodi, viene spostato un numero di chiavi pari a $1/10$ (10 percento).

Sei tu, in qualità di utente, che controlli l'algoritmo di hashing utilizzato per il cluster a più nodi. Ti consigliamo di configurare i client in modo da utilizzare un hashing coerente. Fortunatamente, sono disponibili numerose librerie client Memcached nei linguaggi più comuni in grado di implementare l'hashing coerente. Consulta la documentazione della libreria che utilizzi per verificare se supporta l'hashing coerente e ottenere informazioni su come implementarlo.

Se lavori in Java, PHP o .NET, ti consigliamo di utilizzare una delle librerie client Amazon ElastiCache.

Hashing coerente in Java

Il client Java ElastiCache Memcached è basato su un client Java `spymemcached` open source, con caratteristiche di hashing coerente integrate. La libreria include una classe `KetamaConnectionFactory` che implementa l'hashing coerente. Per impostazione predefinita, l'hashing coerente è disattivato in `spymemcached`.

Per ulteriori informazioni, vedere la documentazione di `KetamaConnectionFactory` all'indirizzo [KetamaConnectionFactory](#).

Hashing coerente in PHP

Il client PHP ElastiCache Memcached è un wrapper intorno alla libreria PHP Memcached integrata. Per impostazione predefinita, l'hashing coerente è disattivato dalla libreria PHP Memcached.

Per attivare l'hashing coerente, utilizza il seguente codice.

```
$m = new Memcached();  
$m->setOption(Memcached::OPT_DISTRIBUTION, Memcached::DISTRIBUTION_CONSISTENT);
```

Oltre al codice precedente, ti consigliamo anche di attivare `memcached.sess_consistent_hash` nel file `php.ini`.

Per ulteriori informazioni, consulta la documentazione relativa alla configurazione di runtime per la libreria PHP Memcached all'indirizzo <http://php.net/manual/en/memcached.configuration.php>. Nota in particolare il parametro `memcached.sess_consistent_hash`.

Hashing coerente in .NET

Il client .NET ElastiCache Memcached è un wrapper intorno a Enyim Memcached. Per impostazione predefinita, l'hashing coerente è attivato dal client Enyim Memcached.

Per ulteriori informazioni, consulta la documentazione di `memcached/locator` all'indirizzo <https://github.com/enyim/EnyimMemcached/wiki/MemcachedClient-Configuration#user-content-memcachedlocator>.

Esempi di client IPv6

Note

Questa sezione si applica ai cluster Memcached progettati autonomamente.

ElastiCache è compatibile con Memcached open source. Ciò significa che i client open source per Memcached che supportano le connessioni IPv6 dovrebbero essere in grado di connettersi a IPv6 abilitati per i cluster Memcached. ElastiCache Inoltre, i seguenti client sono stati convalidati specificamente per funzionare con tutte le configurazioni di tipi di rete supportate:

Di seguito sono riportate le migliori pratiche per interagire con risorse compatibili con IPv6 con librerie client open source di uso comune. ElastiCache È possibile visualizzare [le best practice esistenti con cui interagire per](#) ottenere consigli sulla configurazione dei ElastiCache client per le risorse. ElastiCache Tuttavia, ci sono alcune avvertenze che vale la pena segnalare quando si interagisce con risorse abilitate per IPv6.

Clienti convalidati

Clienti convalidati:

- [AWS ElastiCache Cluster Client Memcached per Php — versione *3.6.2](#)
- [AWS ElastiCache Cluster Client Memcached](#) per Java: master più recente su Github

Configurazione di un protocollo preferito per i cluster dual-stack

Per cluster Memcached puoi controllare il protocollo che verrà utilizzato dai client per connettersi ai nodi del cluster con il parametro IP Discovery. Il parametro IP Discovery può essere impostato su IPv4 o IPv6.

Il parametro IP Discovery controlla il protocollo IP utilizzato nell'output del cluster config get. Il che a sua volta determinerà il protocollo IP utilizzato dai client che supportano l'individuazione automatica ElastiCache per i cluster Memcached.

La modifica di IP Discovery non comporterà alcun tempo di inattività per i client connessi. Tuttavia, la propagazione delle modifiche richiederà tempo.

Monitorare l'output di `getAvailableNodeEndpoints` per Java e per Php monitorare l'output di `getServerList`. Dopo che l'output di queste funzioni segnala gli IP di tutti i nodi del cluster che utilizzano il protocollo aggiornato, la propagazione delle modifiche è terminata.

Esempio di Java

```
MemcachedClient client = new MemcachedClient(new InetSocketAddress("xxxx", 11211));

Class targetProtocolType = Inet6Address.class; // Or Inet4Address.class if you're
switching to IPv4

Set<String> nodes;

do {
```

```

    nodes =
    client.getAvailableNodeEndpoints().stream().map(NodeEndPoint::getIpAddress).collect(Collectors.toList());

    Thread.sleep(1000);
} while (!nodes.stream().allMatch(node -> {
    try {
        return finalTargetProtocolType.isInstance(InetAddress.getByAddress(node));
    } catch (UnknownHostException ignored) {}
    return false;
})));

```

Esempio di Php:

```

$client = new Memcached;
$client->setOption(Memcached::OPT_CLIENT_MODE, Memcached::DYNAMIC_CLIENT_MODE);
$client->addServer("xxxx", 11211);

$nodes = [];
$target_ips_count = 0;
do {
    # The PHP memcached client only updates the server list if the polling interval has
    expired and a
    # command is sent
    $client->get('test');

    $nodes = $client->getServerList();

    sleep(1);
    $target_ips_count = 0;

    // For IPv4 use FILTER_FLAG_IPV4
    $target_ips_count = count(array_filter($nodes, function($node) { return
    filter_var($node["ipaddress"], FILTER_VALIDATE_IP, FILTER_FLAG_IPV6); }));
} while (count($nodes) !== $target_ips_count);

```

Le eventuali connessioni client esistenti create prima dell'aggiornamento di IP Discovery, verranno comunque connesse utilizzando il vecchio protocollo. Tutti i client convalidati si riconnetteranno automaticamente al cluster utilizzando il nuovo protocollo IP una volta rilevate le modifiche nell'output dei comandi di individuazione del cluster. Tuttavia, ciò dipende dall'implementazione del client.

Cluster dual stack abilitati per TLS ElastiCache

Quando TLS è abilitato per ElastiCache i cluster, le funzioni di rilevamento dei cluster restituiscono nomi host anziché IP. I nomi host vengono quindi utilizzati al posto degli IP per connettersi al ElastiCache cluster ed eseguire un handshake TLS. Ciò significa che i client non sono interessati dal parametro Individuazione IP. Per i cluster abilitati per TLS, il parametro Individuazione IP non ha alcun effetto sul protocollo IP preferito. Invece, il protocollo IP utilizzato verrà determinato in base a quello preferito dal client durante la risoluzione dei nomi host DNS.

Client Java

Quando si esegue la connessione da un ambiente Java che supporta sia IPv4 che IPv6, per impostazione predefinita Java preferirà IPv4 rispetto a IPv6 per la compatibilità con le versioni precedenti. Tuttavia, la preferenza del protocollo IP è configurabile tramite gli argomenti JVM. Per preferire IPv4, la JVM accetta `-Djava.net.preferIPv4Stack=true` e per preferire IPv6 imposta `-Djava.net.preferIPv6Stack=true`. Se si imposta `-Djava.net.preferIPv4Stack=true`, significa che la JVM non effettuerà più connessioni IPv6.

Preferenze a livello di host

In generale, se il client o il runtime client non forniscono opzioni di configurazione per l'impostazione di una preferenza del protocollo IP, quando si esegue la risoluzione DNS, il protocollo IP dipenderà dalla configurazione dell'host. Per impostazione predefinita, la maggior parte degli host preferisce IPv6 rispetto a IPv4, ma questa preferenza può essere configurata a livello di host. Ciò influirà su tutte le richieste DNS provenienti da quell'host, non solo su quelle ai cluster. ElastiCache

Host Linux

Per Linux, una preferenza protocollo IP può essere configurata modificando il file `gai.conf`. Il file `gai.conf` è disponibile in `/etc/gai.conf`. Se non è specificato alcun `gai.conf`, uno di esempio deve essere disponibile in `/usr/share/doc/glibc-common-x.xx/gai.conf` che può essere copiato in `/etc/gai.conf`; è quindi necessario rimuovere i commenti dalla configurazione predefinita. Per aggiornare la configurazione in modo da preferire l'IPv4 durante la connessione a un ElastiCache cluster, aggiorna la precedenza per l'intervallo CIDR che comprende gli IP del cluster in modo che sia superiore alla precedenza per le connessioni IPv6 predefinite. Per impostazione predefinita, alle connessioni IPv6 viene assegnata una priorità di 40. Ad esempio, presupponendo che il cluster si trovi in una sottorete con il CIDR `172.31.0.0/16`, la configurazione sottostante porterebbe i client a preferire le connessioni IPv4 a tale cluster.

```
label ::1/128      0
```

```
label ::/0 1
label 2002::/16 2
label ::/96 3
label ::ffff:0:0/96 4
label fec0::/10 5
label fc00::/7 6
label 2001:0::/32 7
label ::ffff:172.31.0.0/112 8
#
# This default differs from the tables given in RFC 3484 by handling
# (now obsolete) site-local IPv6 addresses and Unique Local Addresses.
# The reason for this difference is that these addresses are never
# NATed while IPv4 site-local addresses most probably are. Given
# the precedence of IPv6 over IPv4 (see below) on machines having only
# site-local IPv4 and IPv6 addresses a lookup for a global address would
# see the IPv6 be preferred. The result is a long delay because the
# site-local IPv6 addresses cannot be used while the IPv4 address is
# (at least for the foreseeable future) NATed. We also treat Teredo
# tunnels special.
#
# precedence <mask> <value>
# Add another rule to the RFC 3484 precedence table. See section 2.1
# and 10.3 in RFC 3484. The default is:
#
precedence ::1/128 50
precedence ::/0 40
precedence 2002::/16 30
precedence ::/96 20
precedence ::ffff:0:0/96 10
precedence ::ffff:172.31.0.0/112 100
```

Ulteriori dettagli su `gai.conf` sono disponibili nella [pagina principale di Linux](#)

Host Windows

Il processo per gli host Windows è simile. Per gli host Windows è possibile eseguire `netsh interface ipv6 set prefix CIDR_CONTAINING_CLUSTER_IPS PRECEDENCE LABEL`. L'effetto è identico alla modifica del file `gai.conf` su host Linux.

Ciò aggiornerà le policy di preferenza in modo da preferire le connessioni IPv4 rispetto alle connessioni IPv6 per l'intervallo CIDR specificato. Ad esempio, presupponendo che il cluster si trovi in una sottorete con il CIDR `172.31.0.0/16`, l'esecuzione di `netsh interface ipv6 set`

prefix ::ffff:172.31.0.0:0/112 100 15 genererebbe la seguente tabella di priorità che porterebbe i client a preferire IPv4 durante la connessione al cluster.

```
C:\Users\Administrator>netsh interface ipv6 show prefixpolicies
Querying active state...

Precedence Label Prefix
-----
100 15 ::ffff:172.31.0.0/112
20 4 ::ffff:0:0/96
50 0 ::1/128
40 1 ::/0
30 2 2002::/16
5 5 2001::/32
3 13 fc00::/7
1 11 fec0::/10
1 12 3ffe::/16
1 3 ::/96
```

Comandi Memcached supportati

ElastiCache serverless per Memcached supporta tutti i comandi [memcached](#) open source 1.6 ad eccezione dei seguenti:

- Le connessioni client richiedono TLS, di conseguenza il protocollo UDP non è supportato.
- Il protocollo binario non è supportato, in quanto è ufficialmente [obsoleto](#) in memcached 1.6.
- I comandi GET/GETS sono limitati a 16 KB per evitare potenziali attacchi DoS al server con il recupero di un numero elevato di chiavi.
- Il comando `flush_all` ritardato verrà rifiutato con `CLIENT_ERROR`.
- I comandi che configurano il motore o rivelano informazioni interne sullo stato o sui log del motore non sono supportati, ad esempio:
 - Per il comando `STATS` sono supportati solo `stats` e `stats reset`. Altre varianti restituiscono `ERROR`.
 - `lru / lru_crawler`: modifica delle impostazioni per LRU e crawler LRU.
 - `watch`: controlla i log del server memcached.
 - `verbosity`: configura il livello di log del server.

Strategie di caching

Nel seguente argomento, puoi trovare strategie per popolare e mantenere la cache.

L'implementazione di strategie per il popolamento e la gestione della cache dipende dal tipo di dati che desideri memorizzare e dai modelli di accesso a tali dati. Ad esempio, probabilmente non utilizzerai la stessa strategia per la classifica dei 10 migliori punteggi di gioco in un sito di gaming, o per le notizie più interessanti. Nella parte restante di questa sezione, discuteremo delle strategie più comuni di gestione della cache, dei loro vantaggi e svantaggi.

Argomenti

- [Caricamento lento](#)
- [Write-Through](#)
- [Aggiunta di TTL](#)
- [Argomenti correlati](#)

Caricamento lento

Come dice lo stesso nome, il caricamento lento è una strategia di caching che consente di caricare i dati nella cache solo quando necessario. Funziona come descritto di seguito.

Amazon ElastiCache è un archivio chiave-valore in memoria che si colloca tra la tua applicazione e il datastore (database) ai quali può accedere. Ogni qualvolta la tua applicazione richiama dati, esso invia per prima cosa la richiesta alla cache ElastiCache. Se i dati sono presenti nella cache e sono correnti, ElastiCache restituisce tali dati all'applicazione. Se i dati non sono presenti nella cache o sono scaduti, l'applicazione richiede i dati al Data Store. L'archivio dati restituisce quindi i dati all'applicazione. Successivamente, l'applicazione scrive i dati ricevuti dall'archivio nella cache. In questo modo, può essere recuperato più rapidamente la prossima volta che viene richiesto.

Un'occorrenza della cache si verifica quando i dati sono presenti nella cache e non sono scaduti:

1. L'applicazione richiede i dati alla cache.
2. La cache restituisce i dati all'applicazione.

Una mancato riscontro nella cache si verifica quando i dati non sono presenti nella cache o sono scaduti:

1. L'applicazione richiede i dati alla cache.
2. La cache non dispone dei dati richiesti e restituisce un `null`.
3. L'applicazione richiede e riceve i dati dal database.
4. L'applicazione aggiorna la cache con i nuovi dati.

Vantaggi e svantaggi del caricamento lento

I vantaggi del caricamento lento sono i seguenti:

- Solo i dati richiesti vengono memorizzati nella cache.

Poiché la maggior parte dei dati non viene mai richiesta, il caricamento lento evita di riempire la cache con i dati non richiesti.

- Gli errori dei nodi non sono fatali per l'applicazione.

Se un nodo restituisce un errore e viene sostituito da un nuovo nodo vuoto, l'applicazione continua a funzionare, nonostante l'aumento della latenza. Quando le richieste vengono effettuate al nuovo nodo, ogni mancato riscontro nella cache comporta una query del database. Allo stesso tempo, la copia dei dati viene aggiunta alla cache in modo che le richieste successive vengano richiamate dalla cache.

Gli svantaggi del caricamento pigro sono i seguenti:

- Comporta una penalità per mancato riscontro nella cache. Ogni mancato riscontro nella cache comporta tre passaggi:
 1. Richiesta iniziale dei dati dalla cache
 2. Query del database per i database
 3. Scrittura dei dati sulla cache

Questi mancati riscontri possono causare un notevole ritardo dei dati destinati all'applicazione.

- Dati obsoleti.

Se i dati vengono scritti nella cache solo quando si verifica un mancato riscontro nella cache, i dati nella cache diventano obsoleti. Questo risultato si verifica perché non sono stati aggiornati alla cache quando i dati vengono modificati nel database. Per risolvere questo problema, è possibile utilizzare il [Write-Through](#) e strategie [Aggiunta di TTL](#).

Esempio di pseudocodice di caricamento pigro

Il codice seguente è un esempio di pseudo codice della logica di caricamento lento.

```
// *****  
// function that returns a customer's record.  
// Attempts to retrieve the record from the cache.  
// If it is retrieved, the record is returned to the application.  
// If the record is not retrieved from the cache, it is  
//   retrieved from the database,  
//   added to the cache, and  
//   returned to the application  
// *****  
get_customer(customer_id)  
  
    customer_record = cache.get(customer_id)  
    if (customer_record == null)  
  
        customer_record = db.query("SELECT * FROM Customers WHERE id = {0}",  
customer_id)  
        cache.set(customer_id, customer_record)  
  
    return customer_record
```

Per questo esempio, il codice dell'applicazione che ottiene i dati è il seguente.

```
customer_record = get_customer(12345)
```

Write-Through

La strategia di scrittura contemporanea aggiunge i dati o li aggiorna nella cache ogni qualvolta i dati vengono scritti sul database.

Vantaggi e svantaggi della scrittura contemporanea

I vantaggi di write-through sono i seguenti:

- I dati nella cache non sono mai obsoleti.

Poiché i dati nella cache vengono aggiornati ogni volta che vengono scritti sul database, i dati nella cache sono sempre correnti.

- Penalità di scrittura e penalità di lettura.

Ogni scrittura comporta due passaggi:

1. Una scrittura sulla cache
2. Una scrittura sul database

Che aggiunge latenza al processo. Detto questo, gli utenti finali sono generalmente più tolleranti della latenza durante l'aggiornamento o il richiamo dei dati. Esiste la convinzione che gli aggiornamenti comportino maggior lavoro e richiedano pertanto tempi lunghi.

Gli svantaggi di write-through sono i seguenti:

- Dati mancanti.

Se si esegue un giro verso l'alto di un nuovo nodo, dovuto a un errore del nodo o a un dimensionamento orizzontale, i dati mancanti sono stati visualizzati. Questi dati continuano a mancare fino a quando non vengono aggiunti o aggiornati nel database. È possibile ridurre al minimo questo implementando [caricamento lento](#) con write-through.

- Abbandono della cache.

La maggior parte dei dati non viene mai letta, il che è uno spreco di risorse. Da [aggiunta di un valore durata \(TTL\)](#), è possibile ridurre gli sprechi di spazio.

Esempio di pseudocodice write-through

Il codice seguente è un esempio di pseudo codice della logica di wright-through.

```
// *****  
// function that saves a customer's record.  
// *****  
save_customer(customer_id, values)  
  
    customer_record = db.query("UPDATE Customers WHERE id = {0}", customer_id, values)  
    cache.set(customer_id, customer_record)  
    return success
```

Per questo esempio, il codice dell'applicazione che ottiene i dati è il seguente.

```
save_customer(12345, {"address": "123 Main"})
```

Aggiunta di TTL

Il caricamento lento accetta i dati obsoleti, ma non restituisce errori con i nodi vuoti. La scrittura contemporanea garantisce dati sempre aggiornati ma può restituire un errore con i nodi vuoti e può popolare la cache con dati superflui. Aggiungendo un valore durata (TTL) a ogni scrittura, puoi avere i vantaggi di ogni strategia. Allo stesso tempo, puoi in gran parte evitare di ingombrare la cache con dati extra.

La durata (TTL) è un valore intero che specifica il numero di secondi fino alla scadenza della chiave. Memcached specifica questo valore in secondi. Quando un'applicazione tenta di leggere una chiave scaduta, viene trattata come se la chiave non fosse mai stata trovata. Il database viene interrogato per la chiave e la cache viene aggiornata. Questo approccio non garantisce che un valore non sia obsoleto. Tuttavia, impedisce ai dati di diventare troppo obsoleti e richiede che i valori nella cache vengano occasionalmente aggiornati dal database.

Per ulteriori informazioni, consulta il comando [Memcached set](#).

Esempi di pseudocodice TTL

Il codice seguente è un esempio di pseudo codice della logica di write-through con TTL.

```
// *****  
// function that saves a customer's record.  
// The TTL value of 300 means that the record expires  
//   300 seconds (5 minutes) after the set command  
//   and future reads will have to query the database.  
// *****  
save_customer(customer_id, values)  
  
    customer_record = db.query("UPDATE Customers WHERE id = {0}", customer_id, values)  
    cache.set(customer_id, customer_record, 300)  
  
    return success
```

Il codice seguente è un esempio di pseudo codice della logica di caricamento lento con TTL.

```
// *****  
// function that returns a customer's record.  
// Attempts to retrieve the record from the cache.
```

```
// If it is retrieved, the record is returned to the application.
// If the record is not retrieved from the cache, it is
//   retrieved from the database,
//   added to the cache, and
//   returned to the application.
// The TTL value of 300 means that the record expires
//   300 seconds (5 minutes) after the set command
//   and subsequent reads will have to query the database.
// *****
get_customer(customer_id)

    customer_record = cache.get(customer_id)

    if (customer_record != null)
        if (customer_record.TTL < 300)
            return customer_record          // return the record and exit function

    // do this only if the record did not exist in the cache OR
    //   the TTL was >= 300, i.e., the record in the cache had expired.
    customer_record = db.query("SELECT * FROM Customers WHERE id = {0}", customer_id)
    cache.set(customer_id, customer_record, 300) // update the cache
    return customer_record          // return the newly retrieved record and exit
function
```

Per questo esempio, il codice dell'applicazione che ottiene i dati è il seguente.

```
save_customer(12345, {"address": "123 Main"})
```

```
customer_record = get_customer(12345)
```

Argomenti correlati

- [Datastore in memoria](#)
- [Scelta di un motore e di una versione](#)
- [Scalabilità ElastiCache per Memcached](#)

Gestione del cluster progettato autonomamente

Questa sezione include gli argomenti che ti aiutano a gestire i cluster progettati autonomamente.

Note

Questi argomenti non si applicano a ElastiCache serverless.

Argomenti

- [Gestione della manutenzione](#)
- [Configurazione dei parametri di motore con i gruppi di parametri](#)

Gestione della manutenzione

Ogni cluster ha una finestra di manutenzione settimanale durante la quale vengono applicate le modifiche al sistema. Se non specifichi una finestra di manutenzione preferita quando crei o modifichi un cluster, ElastiCache ne assegna una di 60 minuti all'interno dell'intervallo di manutenzione della tua regione, in un giorno della settimana selezionato casualmente.

La finestra di manutenzione di 60 minuti viene selezionata a caso da un blocco di tempo di 8 ore per regione. La seguente tabella elenca i blocchi temporali per ciascuna regione da cui sono assegnate le finestre di manutenzione predefinite. È possibile scegliere una finestra di manutenzione personalizzata, anche non compresa nel blocco della regione.

Codice regione	Nome della regione	Finestra di manutenzione della regione
ap-northeast-1	Regione Asia Pacifico (Tokyo)	13:00 - 21:00 UTC
ap-northeast-2	Regione Asia Pacifico (Seoul)	12:00 - 20:00 UTC
ap-northeast-3	Regione Asia Pacifico (Osaka-Local)	12:00 - 20:00 UTC
ap-southeast-3	Regione Asia Pacifico (Giacarta)	14:00 - 22:00 UTC
ap-south-1	Regione Asia Pacifico (Mumbai)	17:30-1:30 UTC
ap-southeast-1	Regione Asia Pacifico (Singapore)	14:00 - 22:00 UTC
cn-north-1	Regione Cina (Pechino)	14:00 - 22:00 UTC

Codice regione	Nome della regione	Finestra di manutenzione della regione
cn-northwest-1	Regione Cina (Ningxia)	14:00 - 22:00 UTC
ap-east-1	Regione Asia Pacifico (Hong Kong)	13:00 - 21:00 UTC
ap-southeast-2	Asia Pacifico (Sydney)	12:00 - 20:00 UTC
eu-west-3	Regione UE (Parigi)	23:59 - 07:29 UTC
af-south-1	Regione Africa (Città del Capo)	13:00 - 21:00 UTC
eu-central-1	Regione Europa (Francoforte)	23:00 - 07:00 UTC
eu-west-1	Europa (Irlanda)	22:00 - 06:00 UTC
eu-west-2	Regione Europa (Londra)	23:00 - 07:00 UTC
me-south-1	Regione Medio Oriente (Bahrein)	13:00 - 21:00 UTC
me-central-1	Regione Medio Oriente (Emirati Arabi Uniti)	13:00 - 21:00 UTC
eu-south-1	Regione Europa (Milano)	21:00 - 05:00 UTC
sa-east-1	Regione Sud America (San Paolo)	01:00 - 09:00 UTC
us-east-1	Stati Uniti orientali (Virginia settentrionale)	03:00 - 11:00 UTC
us-east-2	Stati Uniti orientali (Ohio)	04:00–12:00 UTC
us-gov-west-1	Regione AWS GovCloud (US)	06:00 - 14:00 UTC
us-west-1	Regione Stati Uniti occidentali (California settentrionale)	06:00 - 14:00 UTC
us-west-2	Stati Uniti occidentali (Oregon)	06:00 - 14:00 UTC

Modifica della finestra di manutenzione di un cluster

La finestra di manutenzione deve avvenire nel momento dell'utilizzo più basso e pertanto potrebbe essere necessario apportare modifiche di tanto in tanto. Puoi modificare il cluster e specificare un intervallo di tempo di 24 ore al massimo durante il quale si verifichino le attività di manutenzione richieste. Qualsiasi modifica del cluster richiesta, ma posticipata o in sospeso, viene apportata durante questo lasso di tempo.

Note

Se si desidera applicare modifiche al tipo di nodo e/o aggiornamenti del motore immediatamente utilizzando AWS Management Console, seleziona il riquadro **Applica immediatamente**. Altrimenti queste modifiche verranno applicate durante la successiva finestra di manutenzione programmata. Per utilizzare l'API, consulta [modify-replication-group](#) o [modify-cache-cluster](#).

Ulteriori informazioni

Per informazioni sulle finestre di manutenzione e la sostituzione dei nodi, consulta a seguire:

- [Manutenzione ElastiCache](#)—Domande frequenti sulla manutenzione e la sostituzione dei nodi
- [Sostituzione dei nodi](#):-Gestione della sostituzione dei nodi
- [Modifica di un cluster ElastiCache](#) — Modifica della finestra di manutenzione di un cluster

Configurazione dei parametri di motore con i gruppi di parametri

Amazon ElastiCache utilizza i parametri per controllare le proprietà di runtime di nodi e cluster. Le versioni più recenti del prodotto includono in genere parametri aggiuntivi per il support delle nuove funzionalità. Per le tabelle dei parametri, consulta [Parametri Memcached specifici](#).

Come previsto, alcuni valori di parametro, ad esempio `maxmemory`, sono determinati da tipo di nodo e motore. Per una tabella di questi valori di parametro per tipo di nodo, consulta [Parametri specifici del tipo di nodo Memcached](#).

Note

Per un elenco dei parametri specifici Memcached, consultare [Parametri Memcached specifici](#).

Argomenti

- [Gestione dei parametri](#)
- [Livelli dei gruppi di parametri cache](#)
- [Creazione di un gruppo di parametri](#)
- [Elenco di gruppi di parametri per nome](#)
- [Generazione di un elenco di valori di un gruppo di parametri](#)
- [Modifica di un gruppo di parametri](#)
- [Eliminazione di un gruppo di parametri](#)
- [Parametri Memcached specifici](#)

Gestione dei parametri

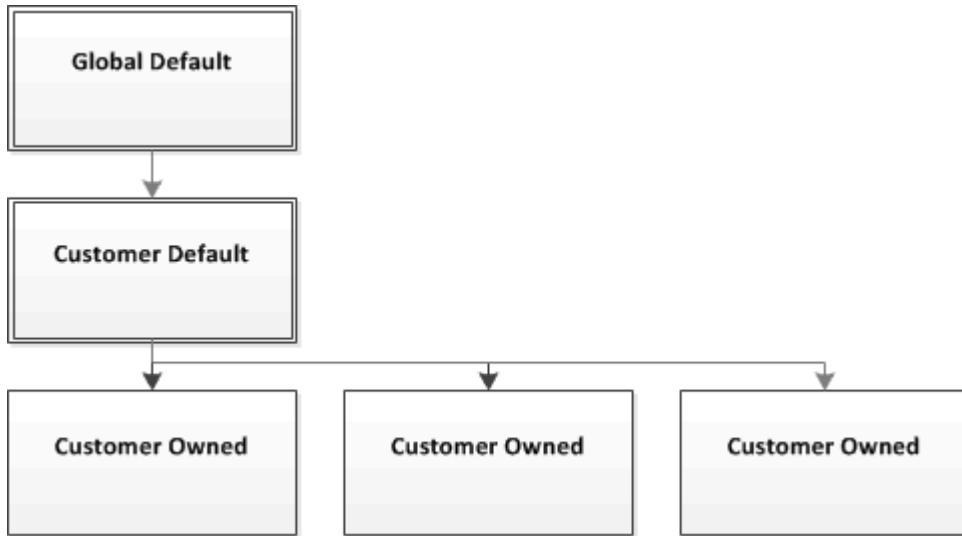
Ai fini di semplificarne la gestione, i parametri sono raggruppati in gruppi di parametri denominati. Un gruppo di parametri rappresenta una combinazione di valori specifici per i parametri passati al software del motore durante l'avvio. Questi valori determinano il comportamento dei processi del motore su ciascun nodo in fase di runtime. I valori dei parametri su un gruppo di parametri specifico si applicano a tutti i nodi associati al gruppo, indipendentemente dal cluster a cui appartengono.

Per ottimizzare le prestazioni del cluster, puoi modificare alcuni valori dei parametri oppure puoi modificare il gruppo di parametri del cluster.

- Non è possibile modificare né eliminare i gruppi di parametri predefiniti. Se hai bisogno di valori dei parametri personalizzati, devi creare un gruppo di parametri personalizzato.
- La famiglia del gruppo di parametri e il cluster che assegna devono essere compatibili. Se il cluster esegue ad esempio Memcached versione 1.4.8, puoi usare solo gruppi di parametri della famiglia Memcached 1.4, predefiniti o personalizzati.
- Se modifichi un gruppo di parametri del cluster, i valori di un parametro modificabile in base a condizioni devono essere gli stessi sia nel gruppo di parametri corrente sia in quello nuovo.
- Quando modifichi i parametri di un cluster, la modifica viene applicata al cluster immediatamente. Ciò è valido se modifichi il gruppo dei parametri del cluster o un valore di parametro nel gruppo dei parametri del cluster. Per determinare quando viene applicata la modifica di un determinato parametro, consulta la colonna relativa all'applicazione delle modifiche nelle tabelle in [Parametri Memcached specifici](#). Per ulteriori informazioni sul riavvio dei nodi del cluster, consulta [Rebooting clusters](#).

Livelli dei gruppi di parametri cache

Amazon ElastiCache ha tre livelli di gruppi di parametri cache come illustrato di seguito.



Livelli dei gruppi di parametri Amazon ElastiCache

Di default globale

Gruppo di parametri root di livello superiore per tutti i clienti di Amazon ElastiCache nella regione.

Gruppo di parametri cache di default globale:

- Riservato per ElastiCache e non disponibile per il cliente.

Di default del cliente

Copia del gruppo di parametri cache di default globale creata per l'uso da parte del cliente.

Il gruppo di parametri cache di default globale:

- Viene creato da ed è di ElastiCache.
- È disponibile per l'uso da parte del cliente come gruppo di parametri cache per i cluster che eseguono una versione del motore supportata dal gruppo di parametri cache.
- Non può essere modificato dal cliente.

Di proprietà del cliente

Copia del gruppo di parametri cache di default globale, un gruppo di parametri cache di proprietà del cliente viene creato ogni volta che il cliente crea un gruppo di parametri cache.

Il gruppo di parametri cache di proprietà del cliente:

- Viene creato dal cliente ed è di sua proprietà.
- Può essere assegnato a un cluster compatibile del cliente.
- Può essere modificato dal cliente per creare un gruppo di parametri cache personalizzato.

Non è possibile modificare tutti i valori dei parametri. Per ulteriori informazioni, consulta [Parametri Memcached specifici](#).

Creazione di un gruppo di parametri

Devi creare un nuovo gruppo di parametri se per uno o più valori di parametri desideri configurare un'impostazione diversa da quella predefinita. Puoi creare un gruppo di parametri utilizzando la console ElastiCache, il AWS CLI o l'API ElastiCache.

Creazione di un gruppo di parametri (Console)

La procedura seguente mostra come creare un gruppo di parametri tramite la console ElastiCache.

Per creare un gruppo di parametri tramite la console ElastiCache

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Per un elenco di tutti i gruppi di parametri disponibili, nel pannello di navigazione a sinistra scegliere Parameter Groups (Gruppi di parametri).
3. Per creare un nuovo gruppo di parametri, scegliere Create Parameter Group (Crea gruppo di parametri).

Viene visualizzata la schermata Create Parameter Group (Crea gruppo di parametri).

4. Nell'elenco Family (Famiglia) scegliere la famiglia del gruppo di parametri che verrà utilizzata come modello per il gruppo di parametri.

La famiglia del gruppo di parametri, ad esempio memcached1.4 , definisce i parametri effettivi nel gruppo di parametri e i relativi valori iniziali. e deve coincidere con la versione e il motore del cluster.

5. Nella casella Name (Nome) digitare un nome univoco per il gruppo di parametri.

Quando si crea un cluster o si modifica un gruppo di parametri del cluster, il gruppo di parametri viene scelto in base al relativo nome. È pertanto consigliabile che il nome sia informativo e identifichi in qualche modo la famiglia del gruppo di parametri.

I vincoli per la denominazione dei gruppi di parametri sono i seguenti:

- Devono iniziare con una lettera ASCII.
 - Può contenere solo lettere ASCII, cifre e trattini ('-').
 - Deve contenere da 1 a 255 caratteri.
 - Non possono contenere due trattini consecutivi.
 - Non posso terminare con un trattino.
6. Nella casella Description (Descrizione) digitare una descrizione per il gruppo di parametri.
 7. Per creare il gruppo di parametri, scegliere Create (Crea).

Per terminare il processo senza creare il gruppo di parametri, scegliere Cancel (Annulla).

8. Quando viene creato, il gruppo di parametri è associato ai valori predefiniti della famiglia. Per modificare i valori predefiniti, è necessario modificare il gruppo di parametri. Per ulteriori informazioni, consulta [Modifica di un gruppo di parametri](#).

Creazione di un gruppo di parametri (AWS CLI)

Per creare un gruppo di parametri tramite AWS CLI, usa il comando `create-cache-parameter-group` con questi parametri.

- `--cache-parameter-group-name` - Nome del gruppo di parametri.

Vincoli per la denominazione dei gruppi di parametri:

- Devono iniziare con una lettera ASCII.
- Può contenere solo lettere ASCII, cifre e trattini ('-').
- Deve contenere da 1 a 255 caratteri.
- Non possono contenere due trattini consecutivi.
- Non posso terminare con un trattino.
- `--cache-parameter-group-family`— Il motore e la famiglia di versioni per il gruppo di parametri.

- `--description` - Una descrizione per la copia del gruppo di parametri del cluster fornita dall'utente.

Example

L'esempio seguente crea un nuovo gruppo di parametri denominato `myMem14`, che utilizza la famiglia `memcached1.4` come modello.

Per Linux, macOS o Unix:

```
aws elasticache create-cache-parameter-group \  
  --cache-parameter-group-name myMem14 \  
  --cache-parameter-group-family memcached1.4 \  
  --description "My first parameter group"
```

Per Windows:

```
aws elasticache create-cache-parameter-group ^  
  --cache-parameter-group-name myMem14 ^  
  --cache-parameter-group-family memcached1.4 ^  
  --description "My first parameter group"
```

L'output di questo comando dovrebbe essere simile a quanto segue:

```
{  
  "CacheParameterGroup": {  
    "CacheParameterGroupName": "myMem14",  
    "CacheParameterGroupFamily": "memcached1.4",  
    "Description": "My first parameter group"  
  }  
}
```

Quando viene creato, il gruppo di parametri è associato ai valori predefiniti della famiglia. Per modificare i valori predefiniti, è necessario modificare il gruppo di parametri. Per ulteriori informazioni, consulta [Modifica di un gruppo di parametri](#).

Per ulteriori informazioni, consulta [create-cache-parameter-group](#).

Creazione di un gruppo di parametri (API ElastiCache)

Per creare un gruppo di parametri tramite l'API ElastiCache, usa l'operazione `CreateCacheParameterGroup` con questi parametri.

- `ParameterGroupName` Nome del gruppo di parametri.

Vincoli per la denominazione dei gruppi di parametri:

- Devono iniziare con una lettera ASCII.
- Può contenere solo lettere ASCII, cifre e trattini ('-').
- Deve contenere da 1 a 255 caratteri.
- Non possono contenere due trattini consecutivi.
- Non posso terminare con un trattino.
- `CacheParameterGroupFamily`— Il motore e la famiglia di versioni per il gruppo di parametri. Ad esempio, `memcached1.4`.
- `Description` - Una descrizione per la copia del gruppo di parametri del cluster fornita dall'utente.

Example

L'esempio seguente crea un nuovo gruppo di parametri denominato `myMem14`, che utilizza la famiglia `memcached1.4` come modello.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=CreateCacheParameterGroup  
&CacheParameterGroupFamily=memcached1.4  
&CacheParameterGroupName=myMem14  
&Description=My%20first%20parameter%20group  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&Version=2015-02-02  
&X-Amz-Credential=<credential>
```

La risposta restituita da tale operazione dovrebbe essere simile a quanto segue:

```
<CreateCacheParameterGroupResponse xmlns="http://elasticache.amazonaws.com/  
doc/2013-06-15/">  
  <CreateCacheParameterGroupResult>  
    <CacheParameterGroup>
```

```
<CacheParameterGroupName>myMem14</CacheParameterGroupName>
<CacheParameterGroupFamily>memcached1.4</CacheParameterGroupFamily>
<Description>My first parameter group</Description>
</CacheParameterGroup>
</CreateCacheParameterGroupResult>
<ResponseMetadata>
  <RequestId>d8465952-af48-11e0-8d36-859edca6f4b8</RequestId>
</ResponseMetadata>
</CreateCacheParameterGroupResponse>
```

Quando viene creato, il gruppo di parametri è associato ai valori predefiniti della famiglia. Per modificare i valori predefiniti, è necessario modificare il gruppo di parametri. Per ulteriori informazioni, consulta [Modifica di un gruppo di parametri](#).

Per ulteriori informazioni, consulta [CreateCacheParameterGroup](#).

Elenco di gruppi di parametri per nome

Puoi elencare i gruppi di parametri utilizzando la console ElastiCache, la AWS CLI o l'API ElastiCache.

Elenco di gruppi di parametri per nome (console)

La procedura seguente mostra come visualizzare un elenco di gruppi di parametri tramite la console ElastiCache.

Per elencare i gruppi di parametri tramite la console ElastiCache

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Per un elenco di tutti i gruppi di parametri disponibili, nel pannello di navigazione a sinistra scegliere Parameter Groups (Gruppi di parametri).

Elenco di gruppi di parametri per nome (AWS CLI)

Per generare un elenco di gruppi di parametri tramite AWS CLI, utilizza il comando `describe-cache-parameter-groups`. Se specifichi un nome del gruppo di parametri, nell'elenco sarà presente solo tale gruppo di parametri. Se non specifichi un nome del gruppo di parametri, nell'elenco saranno presenti fino a `--max-records` gruppi di parametri. In entrambi i casi, saranno indicati nome, famiglia e descrizione del gruppo di parametri.

Example

Il codice di esempio seguente elenca il gruppo di parametri `myMem14`.

Per Linux, macOS o Unix:

```
aws elasticache describe-cache-parameter-groups \  
  --cache-parameter-group-name myMem14
```

Per Windows:

```
aws elasticache describe-cache-parameter-groups ^  
  --cache-parameter-group-name myMem14
```

L'output di questo comando sarà simile al seguente e conterrà il nome, la famiglia e la descrizione del gruppo di parametri.

```
{
  "CacheParameterGroups": [
    {
      "CacheParameterGroupName": "myMem14",
      "CacheParameterGroupFamily": "memcached1.4",
      "Description": "My first parameter group"
    }
  ]
}
```

Example

Il codice di esempio seguente elenca fino a 10 gruppi di parametri.

```
aws elasticache describe-cache-parameter-groups --max-records 10
```

L'output JSON di questo comando sarà simile a questo, elencando per ogni gruppo di parametri il nome, la famiglia, la descrizione e, nel caso di redis5.6, se il gruppo di parametri fa parte di un Global Datastore (isGlobal).

```
{
  "CacheParameterGroups": [
    {
      "CacheParameterGroupName": "custom-redis32",
      "CacheParameterGroupFamily": "redis3.2",
      "Description": "custom parameter group with reserved-memory > 0"
    },
    {
      "CacheParameterGroupName": "default.memcached1.4",
      "CacheParameterGroupFamily": "memcached1.4",
      "Description": "Default parameter group for memcached1.4"
    },
    {
      "CacheParameterGroupName": "default.redis2.6",
      "CacheParameterGroupFamily": "redis2.6",
      "Description": "Default parameter group for redis2.6"
    },
    {
      "CacheParameterGroupName": "default.redis2.8",
```

```

    "CacheParameterGroupFamily": "redis2.8",
    "Description": "Default parameter group for redis2.8"
  },
  {
    "CacheParameterGroupName": "default.redis3.2",
    "CacheParameterGroupFamily": "redis3.2",
    "Description": "Default parameter group for redis3.2"
  },
  {
    "CacheParameterGroupName": "default.redis3.2.cluster.on",
    "CacheParameterGroupFamily": "redis3.2",
    "Description": "Customized default parameter group for redis3.2 with
cluster mode on"
  },
  {
    "CacheParameterGroupName": "default.redis5.6.cluster.on",
    "CacheParameterGroupFamily": "redis5.0",
    "Description": "Customized default parameter group for redis5.6 with
cluster mode on",
    "isGlobal": "yes"
  },
]
}

```

Per ulteriori informazioni, consulta [describe-cache-parameter-groups](#).

Elenco dei gruppi di parametri per nome (API ElastiCache)

Per generare un elenco di gruppi di parametri tramite l'API ElastiCache, utilizza l'operazione `DescribeCacheParameterGroups`. Se specifichi un nome del gruppo di parametri, nell'elenco sarà presente solo tale gruppo di parametri. Se non specifichi un nome del gruppo di parametri, nell'elenco saranno presenti fino a `MaxRecords` gruppi di parametri. In entrambi i casi, saranno indicati nome, famiglia e descrizione del gruppo di parametri.

Example

Il codice di esempio seguente elenca il gruppo di parametri `myMem14`.

```

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheParameterGroups
&CacheParameterGroupName=myMem14
&SignatureVersion=4
&SignatureMethod=HmacSHA256

```

```
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>
```

La risposta restituita da tale operazione sarà simile a quanto segue e conterrà il nome, la famiglia e la descrizione di ciascun gruppo di parametri.

```
<DescribeCacheParameterGroupsResponse xmlns="http://elasticache.amazonaws.com/doc/2013-06-15/">
  <DescribeCacheParameterGroupsResult>
    <CacheParameterGroups>
      <CacheParameterGroup>
        <CacheParameterGroupName>myMem14</CacheParameterGroupName>
        <CacheParameterGroupFamily>memcached1.4</CacheParameterGroupFamily>
        <Description>My custom Memcached 1.4 parameter group</Description>
      </CacheParameterGroup>
    </CacheParameterGroups>
  </DescribeCacheParameterGroupsResult>
  <ResponseMetadata>
    <RequestId>3540cc3d-af48-11e0-97f9-279771c4477e</RequestId>
  </ResponseMetadata>
</DescribeCacheParameterGroupsResponse>
```

Example

Il codice di esempio seguente elenca fino a 10 gruppi di parametri.

```
https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeCacheParameterGroups
&MaxRecords=10
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>
```

La risposta di questa operazione sarà simile a questa, elencando per ogni gruppo di parametri il nome, la famiglia, la descrizione e, nel caso di redis5.6 se il gruppo di parametri appartiene a un Global Datastore (isGlobal).

```
<DescribeCacheParameterGroupsResponse xmlns="http://elasticache.amazonaws.com/doc/2013-06-15/">
```

```
<DescribeCacheParameterGroupsResult>
  <CacheParameterGroups>
    <CacheParameterGroup>
      <CacheParameterGroupName>myRedis28</CacheParameterGroupName>
      <CacheParameterGroupFamily>redis2.8</CacheParameterGroupFamily>
      <Description>My custom Redis 2.8 parameter group</Description>
    </CacheParameterGroup>
    <CacheParameterGroup>
      <CacheParameterGroupName>myMem14</CacheParameterGroupName>
      <CacheParameterGroupFamily>memcached1.4</CacheParameterGroupFamily>
      <Description>My custom Memcached 1.4 parameter group</Description>
    </CacheParameterGroup>
    <CacheParameterGroup>
      <CacheParameterGroupName>myRedis56</CacheParameterGroupName>
      <CacheParameterGroupFamily>redis5.0</CacheParameterGroupFamily>
      <Description>My custom redis 5.6 parameter group</Description>
      <isGlobal>yes</isGlobal>
    </CacheParameterGroup>
  </CacheParameterGroups>
</DescribeCacheParameterGroupsResult>
<ResponseMetadata>
  <RequestId>3540cc3d-af48-11e0-97f9-279771c4477e</RequestId>
</ResponseMetadata>
</DescribeCacheParameterGroupsResponse>
```

Per ulteriori informazioni, consulta [DescribeCacheParameterGroups](#).

Generazione di un elenco di valori di un gruppo di parametri

Puoi creare un elenco di parametri e dei relativi valori di un gruppo di parametri tramite la console ElastiCache, la AWS CLI o l'API ElastiCache.

Generazione di un elenco di valori di un gruppo di parametri (console)

La procedura seguente mostra come creare un elenco dei parametri e dei relativi valori per un gruppo di parametri tramite la console ElastiCache.

Per creare un elenco dei parametri e dei relativi valori in un gruppo di parametri tramite la console ElastiCache.

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Per un elenco di tutti i gruppi di parametri disponibili, nel pannello di navigazione a sinistra scegliere Parameter Groups (Gruppi di parametri).
3. Scegliere il gruppo di parametri per cui si desidera elencare i parametri e i valori selezionando la casella a sinistra del nome del gruppo di parametri.

I parametri e i relativi valori verranno elencati nella parte inferiore dello schermo. A causa dell'elevato numero di parametri, potrebbe essere necessario scorrere verso l'alto e verso il basso per individuare il parametro desiderato.

Generazione di un elenco di valori di un gruppo di parametri (AWS CLI)

Per creare un elenco dei parametri e dei relativi valori in un gruppo di parametri tramite AWS CLI, utilizza il comando `describe-cache-parameters`.

Example

Il seguente codice di esempio consente di creare un elenco di tutti i parametri e dei relativi valori per il gruppo di parametri `myMem14`.

Per Linux, macOS o Unix:

```
aws elasticache describe-cache-parameters \  
  --cache-parameter-group-name myMem14
```

Per Windows:

```
aws elasticache describe-cache-parameters ^  
  --cache-parameter-group-name myMem14
```

Per ulteriori informazioni, consulta [describe-cache-parameters](#).

Generazione di un elenco di valori di un gruppo di parametri (API ElastiCache)

Per creare un elenco dei parametri e dei relativi valori in un gruppo di parametri tramite l'API ElastiCache, utilizza l'operazione DescribeCacheParameters.

Example

Il seguente codice di esempio consente di creare un elenco di tutti i parametri per il gruppo di parametri myMem14.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=DescribeCacheParameters  
  &CacheParameterGroupName=myMem14  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &Version=2015-02-02  
  &X-Amz-Credential=<credential>
```

La risposta restituita da tale operazione dovrebbe essere simile a quanto segue. In questo caso, la risposta è stata troncata.

```
<DescribeCacheParametersResponse xmlns="http://elasticache.amazonaws.com/  
doc/2013-06-15/">  
  <DescribeCacheParametersResult>  
    <CacheClusterClassSpecificParameters>  
      <CacheNodeTypeSpecificParameter>  
        <DataType>integer</DataType>  
        <Source>system</Source>  
        <IsModifiable>>false</IsModifiable>  
        <Description>The maximum configurable amount of memory to use to store items,  
in megabytes.</Description>  
        <CacheNodeTypeSpecificValues>  
          <CacheNodeTypeSpecificValue>  
            <Value>1000</Value>  
          <CacheClusterClass>cache.c1.medium</CacheClusterClass>
```

```
</CacheNodeTypeSpecificValue>
<CacheNodeTypeSpecificValue>
  <Value>6000</Value>
  <CacheClusterClass>cache.c1.xlarge</CacheClusterClass>
</CacheNodeTypeSpecificValue>
<CacheNodeTypeSpecificValue>
  <Value>7100</Value>
  <CacheClusterClass>cache.m1.large</CacheClusterClass>
</CacheNodeTypeSpecificValue>
<CacheNodeTypeSpecificValue>
  <Value>1300</Value>
  <CacheClusterClass>cache.m1.small</CacheClusterClass>
</CacheNodeTypeSpecificValue>
```

...output omitted...

```
</CacheClusterClassSpecificParameters>
</DescribeCacheParametersResult>
<ResponseMetadata>
  <RequestId>6d355589-af49-11e0-97f9-279771c4477e</RequestId>
</ResponseMetadata>
</DescribeCacheParametersResponse>
```

Per ulteriori informazioni, consulta [DescribeCacheParameters](#).

Modifica di un gruppo di parametri

Important

Non è consentito modificare un gruppo di parametri di default.

Non puoi modificare alcuni valori di parametri in un gruppo di parametri. Tali valori di parametri sono applicati ai cluster associati al gruppo di parametri. Per ulteriori informazioni su quando una modifica dei valori di parametri viene applicata a un gruppo di parametri, consulta [Parametri Memcached specifici](#).

Modifica di un gruppo di parametri (console)

La procedura seguente mostra come modificare il valore del parametro `binding_protocol` tramite la console ElastiCache. Puoi usare la stessa procedura per modificare il valore di qualsiasi parametro.

Per modificare il valore di un parametro tramite la console ElastiCache

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Per un elenco di tutti i gruppi di parametri disponibili, nel pannello di navigazione a sinistra scegliere Parameter Groups (Gruppi di parametri).
3. Scegliere il gruppo di parametri che si desidera modificare selezionando la casella a sinistra del nome del gruppo di parametri.

I parametri del gruppo di parametri verranno elencati nella parte inferiore dello schermo. Potrebbe essere necessario scorrere l'elenco per visualizzare tutti i parametri.

4. Per modificare uno o più parametri, scegliere Edit Parameters (Modifica parametri).
5. Nella schermata Edit Parameter Group: (Modifica il gruppo di parametri:) scorrere utilizzando le frecce sinistra e destra fino al parametro `binding_protocol`, quindi digitare `ascii` nella colonna Value (Valore).
6. Nella schermata Edit Parameter Group: (Modifica il gruppo di parametri:) scorrere utilizzando le frecce sinistra e destra fino al parametro `cluster-enabled`, quindi digitare `yes` nella colonna Value (Valore).
7. Seleziona Salva modifiche.
8. Per trovare il nome del parametro modificato, consultare [Parametri Memcached specifici](#). Se le modifiche apportate al parametro vengono applicate dopo il riavvio, riavvia ogni cluster che utilizza il gruppo di parametri. Per ulteriori informazioni, consulta [Rebooting clusters](#).

Modifica di un gruppo di parametri (AWS CLI)

Per modificare il valore di un parametro tramite AWS CLI, utilizza il comando `modify-cache-parameter-group`.

Example

Per individuare il nome e i valori consentiti del parametro da modificare, consulta [Parametri Memcached specifici](#)

Il seguente codice di esempio imposta il valore di due parametri, `chunk_size` e `chunk_size_growth_fact`, nel gruppo di parametri `myMem14`.

Per Linux, macOS o Unix:

```
aws elasticache modify-cache-parameter-group \  
  --cache-parameter-group-name myMem14 \  
  --parameter-name-values \  
    ParameterName=chunk_size,ParameterValue=96 \  
    ParameterName=chunk_size_growth_fact,ParameterValue=1.5
```

Per Windows:

```
aws elasticache modify-cache-parameter-group ^  
  --cache-parameter-group-name myMem14 ^  
  --parameter-name-values ^  
    ParameterName=chunk_size,ParameterValue=96 ^  
    ParameterName=chunk_size_growth_fact,ParameterValue=1.5
```

L'aspetto dell'output di questo comando sarà simile al seguente.

```
{  
  "CacheParameterGroupName": "myMem14"  
}
```

Per ulteriori informazioni, consulta [modify-cache-parameter-group](#).

Modifica di un gruppo di parametri (API ElastiCache)

Per modificare i valori dei parametri di un gruppo di parametri tramite l'API ElastiCache, utilizza l'operazione `ModifyCacheParameterGroup`.

Example

Per individuare il nome e i valori consentiti del parametro da modificare, consulta [Parametri Memcached specifici](#)

Il seguente codice di esempio imposta il valore di due parametri, `chunk_size` e `chunk_size_growth_fact`, nel gruppo di parametri `myMem14`.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=ModifyCacheParameterGroup  
  &CacheParameterGroupName=myMem14  
  &ParameterNameValues.member.1.ParameterName=chunk_size  
  &ParameterNameValues.member.1.ParameterValue=96  
  &ParameterNameValues.member.2.ParameterName=chunk_size_growth_fact
```

```
&ParameterNameValues.member.2.ParameterValue=1.5  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Timestamp=20150202T192317Z  
&Version=2015-02-02  
&X-Amz-Credential=<credential>
```

Per ulteriori informazioni, consulta [ModifyCacheParameterGroup](#).

Eliminazione di un gruppo di parametri

Puoi creare un gruppo di parametri utilizzando la console ElastiCache, la AWS CLI o l'API ElastiCache.

Un gruppo di parametri non può essere eliminato se è associato a cluster. Non è inoltre possibile eliminare i gruppi di parametri predefiniti.

Eliminazione di un gruppo di parametri (console)

La procedura seguente mostra come eliminare un gruppo di parametri tramite la console ElastiCache.

Per eliminare un gruppo di parametri tramite la console ElastiCache

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Per un elenco di tutti i gruppi di parametri disponibili, nel pannello di navigazione a sinistra scegliere Parameter Groups (Gruppi di parametri).
3. Scegliere i gruppi di parametri che si desidera eliminare selezionando la casella a sinistra del nome del gruppo di parametri.

Il pulsante Delete (Elimina) diventa attivo.

4. Scegliere Delete (Elimina).

Verrà visualizzata la schermata di conferma Delete Parameter Groups (Elimina gruppi di parametri).

5. Per eliminare i gruppi di parametri, nella schermata di conferma Delete Parameter Groups (Elimina gruppi di parametri) scegliere Delete (Elimina).

Per mantenere i gruppi di parametri, scegliere Cancel (Annulla).

Eliminazione di un gruppo di parametri (AWS CLI)

Per eliminare un gruppo di parametri tramite AWS CLI, utilizza il comando `delete-cache-parameter-group`. Per il gruppo di parametri da eliminare, il gruppo di parametri specificato da `--cache-parameter-group-name` non può essere associato ad alcun cluster, né può essere un gruppo di parametri di default.

Il codice di esempio seguente elimina il gruppo di parametri `myMem14`.

Example

Per Linux, macOS o Unix:

```
aws elasticache delete-cache-parameter-group \  
  --cache-parameter-group-name myMem14
```

Per Windows:

```
aws elasticache delete-cache-parameter-group ^  
  --cache-parameter-group-name myMem14
```

Per ulteriori informazioni, consulta [delete-cache-parameter-group](#).

Eliminazione di un gruppo di parametri (API ElastiCache)

Per eliminare un gruppo di parametri tramite l'API ElastiCache, utilizza l'operazione `DeleteCacheParameterGroup`. Per il gruppo di parametri da eliminare, il gruppo di parametri specificato da `CacheParameterGroupName` non può essere associato ad alcun cluster, né può essere un gruppo di parametri di default.

Example

Il codice di esempio seguente elimina il gruppo di parametri `myMem14`.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=DeleteCacheParameterGroup  
  &CacheParameterGroupName=myMem14  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20150202T192317Z  
  &Version=2015-02-02  
  &X-Amz-Credential=<credential>
```

Per ulteriori informazioni, consulta [DeleteCacheParameterGroup](#).

Parametri Memcached specifici

Se non specifichi un gruppo di parametri per il cluster Memcached, verrà utilizzato un gruppo di parametri di default appropriato per la versione del motore. Non è possibile modificare nessun valore all'interno del gruppo di parametri di default. Tuttavia, è possibile creare un gruppo di parametri personalizzato e assegnarlo al cluster in qualsiasi momento. Per ulteriori informazioni, consulta [Creazione di un gruppo di parametri](#).

Argomenti

- [Modifiche di Memcached 1.6.17](#)
- [Parametri di Memcached 1.6.6 aggiunti](#)
- [Modifiche dei parametri di Memcached 1.5.10](#)
- [Parametri di Memcached 1.4.34 aggiunti](#)
- [Parametri di Memcached 1.4.33 aggiunti](#)
- [Parametri di Memcached 1.4.24 aggiunti](#)
- [Parametri di Memcached 1.4.14 aggiunti](#)
- [Parametri di Memcached 1.4.5 supportati](#)
- [Sovraccarico delle connessioni Memcached](#)
- [Parametri specifici del tipo di nodo Memcached](#)

Modifiche di Memcached 1.6.17

A partire da Memcached 1.6.17 non sono più supportati questi comandi amministrativi: `lru_crawler`, `lru` e `slabs`. Con queste modifiche non sarai in grado di abilitare/disabilitare `lru_crawler` in fase di runtime usando i comandi. Abilita/disabilita `lru_crawler` modificando il tuo gruppo di parametri personalizzato.

Parametri di Memcached 1.6.6 aggiunti


Per Memcached 1.6.6, non sono supportati parametri aggiuntivi.

Famiglia del gruppo di parametri: `memcached1.6`

Modifiche dei parametri di Memcached 1.5.10

Per Memcached 1.5.10, sono supportati i seguenti parametri aggiuntivi.

Famiglia del gruppo di parametri: `memcached1.5`

Nome	Informazioni	Descrizione
no_modern	<p>Impostazione predefinita: 1</p> <p>Tipo: booleano</p> <p>Modificabile: sì</p> <p>Valori consentiti: 0,1</p> <p>Modificazioni hanno effetto: All'avvio</p>	<p>Un alias per disabilitare i comandi <code>slab_reassign</code> , <code>slab_automove</code> , <code>lru_crawler</code> , <code>lru_maintainer</code> , <code>maxconns_fast</code> . No modern imposta anche l'algoritmo hash su <code>jenkins</code> e consente l'inlining di ASCII VALUE. Applicabile a memcached 1.5 e versioni successive. Per tornare al moderno, è necessario disabilitare questo parametro e riavviare , che abilita automaticamente <code>slab_reassign</code> , <code>slab_automove</code> , <code>lru_crawler</code> , <code>lru_maintainer</code> e <code>maxconns_fast</code> .</p> <div data-bbox="1008 1171 1511 1885"><p> Note</p><p>Il valore di configurazione di default per questo parametro è stato modificato da 0 a 1 a partire dal 20 agosto 2021. Il valore di default aggiornato verrà automaticamente rilevato dai nuovi utenti di Elasticache per ciascuna regione dopo il 20 agosto 2021. Gli utenti di ElastiCache esistenti nelle regioni prima del</p></div>

Nome	Informazioni	Descrizione
		20 agosto 2021 devono modificare manualmente i gruppi di parametri personalizzati per poter rilevare questa nuova modifica.
<code>inline_ascii_resp</code>	Impostazione Predefinita: 0 Tipo: booleano Modificabile: sì Valori consentiti: 0,1 Modificazioni hanno effetto: Al lancio	Archivia i numeri dalla risposta VALUE in una voce usando fino a 24 byte. Lieve rallentamento dei set ASCII get, faster.

Per Memcached 1.5.10, i seguenti parametri sono stati rimossi.

Nome	Informazioni	Descrizione
<code>expirezero_does_no_t_evict</code>	Impostazione Predefinita: 0 Tipo: booleano Modificabile: sì Valori consentiti: 0,1 Modificazioni hanno effetto: Al lancio	Non più supportati in questa versione.
<code>modern</code>	Impostazione predefinita: 1	

Nome	Informazioni	Descrizione
	Tipo: booleano Modificabile: Sì (richiede il riavvio, se impostato su <code>no_modern</code>) Valori consentiti: 0,1 Modificazioni hanno effetto: Al lancio	Non più supportati in questa versione. A partire da questa versione <code>no-modern</code> è abilitato per impostazione predefinita a ogni avvio o riavvio.

Parametri di Memcached 1.4.34 aggiunti

Per Memcached 1.4.34, non sono supportati parametri aggiuntivi.

Famiglia del gruppo di parametri: `memcached1.4`

Parametri di Memcached 1.4.33 aggiunti

Per Memcached 1.4.33, sono supportati i seguenti parametri aggiuntivi.

Famiglia del gruppo di parametri: `memcached1.4`

Nome	Informazioni	Descrizione
<code>modern</code>	Impostazione predefinita: abilitato Tipo: booleano Modificabile: sì Modificazioni hanno effetto: Al lancio	Un alias per più funzionalità. L'abilitazione di <code>modern</code> equivale all'attivazione dei seguenti comandi e all'utilizzo dell'algoritmo hash murmur3: <code>slab_reassign</code> , <code>slab_automove</code> , <code>lru_crawler</code> , <code>lru_maintainer</code> , <code>maxconns_fast</code> e <code>hash_algorithm=murmur3</code> .

Nome	Informazioni	Descrizione
<code>watch</code>	<p>Impostazione predefinita: abilitato</p> <p>Tipo: booleano</p> <p>Modificabile: sì</p> <p>Le modifiche diventano effettive: immediatamente</p> <p>I log possono essere eliminati se l'utente raggiunge i limiti relativi a <code>watcher_logbuf_size</code> e <code>worker_logbuf_size</code>.</p>	<p>Recupero, espulsione o mutazione di log. Ad esempio, quando un utente attiva <code>watch</code>, può visualizzare i log quando si verifica <code>get</code>, <code>set</code>, <code>delete</code> o <code>update</code>.</p>
<code>idle_timeout</code>	<p>Impostazione predefinita: 0 (disabilitato)</p> <p>Tipo: Integer</p> <p>Modificabile: sì</p> <p>Le modifiche diventano effettive: Al lancio</p>	<p>Il numero minimo di secondi durante i quali un client può essere inattivo prima che ne venga richiesta la chiusura. Intervallo di valori: da 0 a 86400.</p>

Nome	Informazioni	Descrizione
<code>track_sizes</code>	<p>Impostazione predefinita: disabilitato</p> <p>Tipo: booleano</p> <p>Modificabile: sì</p> <p>Le modifiche diventano effettive: Al lancio</p>	<p>Mostra le dimensioni utilizzate da ciascun gruppo di slab.</p> <p>L'abilitazione di <code>track_sizes</code> consente di eseguire <code>stats sizes</code> senza dover eseguire <code>stats sizes_enable</code>.</p>
<code>watcher_logbuf_size</code>	<p>Impostazione predefinita: 256 (KB)</p> <p>Tipo: Integer</p> <p>Modificabile: sì</p> <p>Le modifiche diventano effettive: Al lancio</p>	<p>Il comando <code>watch</code> abilita la registrazione del flusso per Memcached. <code>watch</code> può tuttavia eliminare i log se il tasso di espulsioni, mutazioni o recuperi è sufficientemente elevato per riempire il buffer di registrazione. In tali situazioni, gli utenti possono aumentare le dimensioni del buffer in modo da ridurre la possibile perdita di log.</p>
<code>worker_logbuf_size</code>	<p>Impostazione predefinita: 64 (KB)</p> <p>Tipo: Integer</p> <p>Modificabile: sì</p> <p>Le modifiche diventano effettive: Al lancio</p>	<p>Il comando <code>watch</code> abilita la registrazione del flusso per Memcached. <code>watch</code> può tuttavia eliminare i log se il tasso di espulsioni, mutazioni o recuperi è sufficientemente elevato per riempire il buffer di registrazione. In tali situazioni, gli utenti possono aumentare le dimensioni del buffer in modo da ridurre la possibile perdita di log.</p>

Nome	Informazioni	Descrizione
<code>slab_chunk_max</code>	<p>Impostazione predefinita: 524288 (byte)</p> <p>Tipo: Integer</p> <p>Modificabile: sì</p> <p>Le modifiche diventano effettive: Al lancio</p>	<p>Specifica le dimensioni massime di uno slab. Impostando una dimensione inferiore di slab viene utilizzata una memoria più efficiente. Gli elementi più grandi di <code>slab_chunk_max</code> sono suddivisi in più slab.</p>
<code>lru_crawler metadump [all 1 2 3]</code>	<p>Impostazione predefinita: disabilitato</p> <p>Tipo: booleano</p> <p>Modificabile: sì</p> <p>Le modifiche diventano effettive: immediatamente</p>	<p>Se <code>lru_crawler</code> è abilitato questo comando effettua il dump di tutte le chiavi.</p> <p><code>all 1 2 3</code> : tutti gli slab o specificare un determinato numero di slab</p>



Parametri di Memcached 1.4.24 aggiunti


Per Memcached 1.4.24, sono supportati i seguenti parametri aggiuntivi.

Famiglia del gruppo di parametri: `memcached1.4`

Nome	Informazioni	Descrizione
<code>disable_flush_all</code>	<p>Impostazione predefinita: 0 (disabilitato)</p> <p>Tipo: booleano</p> <p>Modificabile: sì</p>	<p>Aggiungere il parametro (-F) per disabilitare <code>flush_all</code>. Utile se non desideri poter eseguire uno svuotamento completo sulle istanze di produzione.</p>

Nome	Informazioni	Descrizione
	Modificazioni hanno effetto: Al lancio	Valori: 0, 1 (l'utente può utilizzare il parametro <code>flush_all</code> quando il valore è 0).
<code>hash_algorithm</code>	Impostazione predefinita: <code>jenkins</code> Tipo: stringa Modificabile: sì Modificazioni hanno effetto: Al lancio	L'algoritmo hash da utilizzare. Valori consentiti: <code>murmur3</code> e <code>jenkins</code> .

Nome	Informazioni	Descrizione
<code>lru_crawler</code>	<p>Impostazione predefinita: 0 (disabilitato)</p> <p>Tipo: booleano</p> <p>Modificabile: sì</p> <p>Le modifiche hanno effetto: Dopo il riavvio</p> <div data-bbox="651 682 971 1430"><p> Note</p><p>Puoi abilitare temporaneamente <code>lru_crawler</code> in fase di runtime dalla riga di comando. Per ulteriori informazioni, consulta la colonna Descrizione.</p></div>	<p>Pulisce le classi di slab degli elementi scaduti. Si tratta di un processo a basso impatto che viene eseguito in background. Attualmente richiede l'avvio di un crawling tramite l'utilizzo di un comando manuale.</p> <p>Per un'abilitazione temporanea, esegui il comando <code>lru_crawler enable</code> alla riga di comando.</p> <p><code>lru_crawler 1,3,5</code> esegue il crawling delle classi slab 1, 3 e 5 per cercare elementi scaduti da aggiungere all'elenco libero.</p> <p>Valori: 0,1</p> <div data-bbox="1008 1098 1507 1799"><p> Note</p><p>L'abilitazione di <code>lru_crawler</code> alla riga di comando abilita il crawler finché non viene disabilitato alla riga di comando o al successivo riavvio. Per abilitarlo in modo permanente, devi modificarne il valore del parametro. Per ulteriori informazioni, consulta Modifica di un gruppo di parametri.</p></div>

Nome	Informazioni	Descrizione
<code>lru_maintainer</code>	Impostazione predefinita: 0 (disabilitato) Tipo: booleano Modificabile: sì Modificazioni hanno effetto: Al lancio	Un thread in background che distribuisce gli elementi utilizzati meno di recente man mano che si raggiungono le capacità. Valori: 0, 1.
<code>expirezero_does_not_evict</code>	Impostazione predefinita: 0 (disabilitato) Tipo: booleano Modificabile: sì Modificazioni hanno effetto: Al lancio	Quando utilizzato con <code>lru_maintainer</code> , rende gli elementi non soggetti a espulsione con un tempo di scadenza pari a 0. <div style="border: 1px solid #f08080; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p> Warning</p> <p>Ciò consente di togliere spazio di memoria disponibile per gli elementi soggetti a espulsione.</p> </div> <p>Può essere impostato in modo da ignorare <code>lru_maintainer</code>.</p>

Parametri di Memcached 1.4.14 aggiunti

Per Memcached 1.4.14, sono supportati i seguenti parametri aggiuntivi.

Famiglia del gruppo di parametri: `memcached1.4`

Parametri aggiunti in Memcached 1.4.14

Nome	Descrizione
<code>config_max</code>	Il numero massimo di voci di configurazione ElastiCache.
<code>config_size_max</code>	Dimensione massima delle voci di configurazione, in byte.
<code>hashpower_init</code>	La dimensione iniziale della tabella hash ElastiCache, espressa come potenza di due. L'impostazione 16 (2^{16}) o 65536 chiavi.

Nome	Descrizione
maxconns_fast	<p>Modifica il modo in cui vengono gestite le nuove richieste di connessione quando viene raggiunto il limite massimo di connessioni. Se questo parametro è impostato su 0 (zero), nuove connessioni vengono aggiunte alla coda di backlog e verrà attesa la chiusura delle altre connessioni. Se il parametro è impostato su 1, ElastiCache invia un errore al client e chiude immediatamente la connessione.</p>

Nome	Descrizione
s1ab_automove	<p>Regola l'algoritmo slab automove: se questo parametro è impostato su 0 (zero), l'algoritmo è disabilitato. Se è impostato su 1, ElastiCache utilizza un lento approccio tradizionale per lo spostamento automatico degli slab. Se è impostato su 2, ElastiCache sposta improvvisamente gli slab ogni volta che si verifica un'espulsione. Questa modalità è consigliabile esclusivamente a scopo di test.</p>

Nome	Descrizione
<code>slab_reassign</code>	Abilitare o disabilitare la riassegnazione dello slab. Se questo parametro è impostato su 1, puoi utilizzare il comando di "riassegnazione slab" (<code>slab_reassign</code>) per riassegnare manualmente la memoria.

Parametri di Memcached 1.4.5 supportati

Famiglia del gruppo di parametri: `memcached1.4`

Per Memcached 1.4.5, sono supportati i seguenti parametri.

Parametri aggiunti in Memcached 1.4.5

Nome	Informazioni	Descrizione
<code>backlog_queue_limit</code>	Impostazione predefinita: 1024 Tipo: Integer Modificabile: no	Limite della coda di backlog.

Nome	Informazioni	Descrizione
<code>binding_protocol</code>	<p>Impostazione predefinita: <code>auto</code></p> <p>Tipo: stringa</p> <p>Modificabile: sì</p> <p>Le modifiche hanno effetto: Dopo il riavvio</p>	<p>Il protocollo di binding.</p> <p>Valori consentiti: <code>ascii</code> e <code>auto</code>.</p> <p>Per ulteriori informazioni sulla modifica del valore di <code>binding_protocol</code>, consulta Modifica di un gruppo di parametri.</p>
<code>cas_disabled</code>	<p>Impostazione predefinita: <code>0(false)</code></p> <p>Tipo: Booleano</p> <p>Modificabile: sì</p> <p>Le modifiche hanno effetto: Dopo il riavvio</p>	<p>Se <code>1 (true)</code>, le operazioni di controllo e impostazione (CAS) verranno disabilitate e gli elementi archiviati utilizzeranno 8 byte in meno rispetto a quando CAS è abilitato.</p>
<code>chunk_size</code>	<p>Impostazione predefinita: <code>48</code></p> <p>Tipo: Integer</p> <p>Modificabile: sì</p> <p>Le modifiche hanno effetto: Dopo il riavvio</p>	<p>La quantità minima, in byte, di spazio da allocare per chiave, valore e flag dell'elemento più piccolo.</p>
<code>chunk_size_growth_factor</code>	<p>Impostazione predefinita: <code>1.25</code></p> <p>Tipo: Float</p> <p>Modificabile: sì</p> <p>Le modifiche hanno effetto: Dopo il riavvio</p>	<p>Il fattore di crescita che controlla le dimensioni di ciascun successivo blocco di Memcached. Ciascun blocco sarà maggiore del precedente e in base al valore di <code>chunk_size_growth_factor</code>.</p>

Nome	Informazioni	Descrizione
<code>error_on_memory_exhausted</code>	Impostazione predefinita: 0(false) Tipo: Booleano Modificabile: sì Le modifiche hanno effetto: Dopo il riavvio	Se 1 (true), quando si esaurisce la memoria per archiviare gli elementi, Memcached restituirà un errore anziché espellere gli elementi.
<code>large_memory_pages</code>	Impostazione predefinita: 0(false) Tipo: Booleano Modificabile: no	Se 1 (true), ElastiCache tenterà di utilizzare pagine di memoria di grandi dimensioni.
<code>lock_down_paged_memory</code>	Impostazione predefinita: 0(false) Tipo: Booleano Modificabile: no	Se 1 (true), ElastiCache bloccherà tutta la memoria paginata.
<code>max_item_size</code>	Impostazione predefinita: 1048576 Tipo: Integer Modificabile: sì Le modifiche hanno effetto: Dopo il riavvio	La dimensione, in byte, dell'elemento più grande che può essere archiviato nel cluster.

Nome	Informazioni	Descrizione
<code>max_simultaneous_connections</code>	Impostazione predefinita: 65000 Tipo: Integer Modificabile: no	Il numero massimo di connessioni simultanee.
<code>maximize_core_file_limit</code>	Impostazione predefinita: 0(false) Tipo: Booleano Modificabile: Le modifiche hanno effetto: Dopo il riavvio	Se 1 (true), ElastiCache aumenterà il limite del file core.
<code>memcached_connections_overhead</code>	Impostazione predefinita: 100 Tipo: Integer Modificabile: sì Le modifiche hanno effetto: Dopo il riavvio	La quantità di memoria da riservare per le connessioni Memcached e altro sovraccarico vario. Per informazioni su questo parametro , consulta Sovraccarico delle connessioni Memcached .
<code>requests_per_event</code>	Impostazione predefinita: 20 Tipo: Integer Modificabile: no	Il numero massimo di richieste per evento per una connessione specifica. Questo limite è necessario per evitare un uso eccessivo delle risorse.

Sovraccarico delle connessioni Memcached

Su ciascun nodo, la memoria resa disponibile per l'archiviazione degli elementi è la memoria totale disponibile sul nodo, archiviata nel parametro `max_cache_memory`, meno la memoria utilizzata per le connessioni e altro sovraccarico, archiviata nel parametro `memcached_connections_overhead`. Un nodo di tipo `cache.m1.small`

ha ad esempio un valore di `max_cache_memory` pari a 1300 MB. Con il valore `memcached_connections_overhead` di default di 100 MB, il processo Memcached avrà 1200 MB disponibili per archiviare gli elementi.

I valori predefiniti per il parametro `memcached_connections_overhead` sono sufficienti nella maggior parte dei casi. La quantità di allocazione richiesta per il sovraccarico delle connessioni può tuttavia variare in base a più fattori, tra cui tasso di richiesta, dimensione del payload e numero di connessioni.

Puoi modificare il valore di `memcached_connections_overhead` per soddisfare al meglio le esigenze della tua applicazione. Aumentando ad esempio il valore del parametro `memcached_connections_overhead` otterrai la riduzione della quantità di memoria disponibile per l'archiviazione degli elementi e verrà reso disponibile un buffer di dimensioni maggiori per il sovraccarico delle connessioni. Riducendo il valore del parametro `memcached_connections_overhead` avrai a disposizione una maggiore quantità di memoria per archiviare gli elementi, sebbene ciò aumenti il rischio di utilizzo di swap e di riduzione delle prestazioni. Se riscontri utilizzo dello swap e riduzione delle prestazioni, prova ad aumentare il valore del parametro `memcached_connections_overhead`.

Important

Per il tipo di nodo `cache.t1.micro`, il valore di `memcached_connections_overhead` viene determinato come segue:

- Se il cluster utilizza il gruppo di parametri di default, ElastiCache imposterà il valore di `memcached_connections_overhead` su 13 MB.
- Se il cluster utilizza un gruppo di parametri che tu hai creato, puoi impostare `memcached_connections_overhead` su un valore di tua scelta.

Parametri specifici del tipo di nodo Memcached

Sebbene la maggior parte dei parametri abbia un valore singolo, alcuni parametri hanno diversi valori in base al tipo di nodo utilizzato. La tabella seguente mostra i valori predefiniti dei parametri `max_cache_memory` e `num_threads` per ciascun tipo di nodo. I valori di questi parametri non possono essere modificati.

Tipo di nodo	max_cache_memory (in megabyte)	num_threads
cache.t1.micro	213	1
cache.t2.micro	555	1
cache.t2.small	1588	1
cache.t2.medium	3301	2
cache.t3.micro	512	2
cache.t3.small	1402	2
cache.t3.medium	3364	2
cache.t4g.micro	512	2
cache.t4g.small	1402	2
cache.t4g.medium	3164	2
cache.m1.small	1301	1
cache.m1.medium	3350	1
cache.m1.large	7100	2
cache.m1.xlarge	14600	4
cache.m2.xlarge	33800	2
cache.m2.2xlarge	30412	4
cache.m2,4xlarge	68000	16
cache.m3.medium	2850	1
cache.m3.large	6200	2
cache.m3.xlarge	13600	4

Tipo di nodo	max_cache_memory (in megabyte)	num_threads
cache.m3,2xlarge	28600	8
cache.m4.large	6573	2
cache.m4.xlarge	11496	4
cache.m4.2xlarge	30412	8
cache.m4.4xlarge	62234	16
cache.m4.10xlarge	158355	40
cache.m5.large	6537	2
cache.m5.xlarge	13248	4
cache.m5.2xlarge	26671	8
cache.m5.4xlarge	53516	16
cache.m5.12xlarge	160900	48
cache.m5.24xlarge	321865	96
cache.m6g.large	6537	2
cache.m6g.xlarge	13248	4
cache.m6g.2xlarge	26671	8
cache.m6g.4xlarge	53516	16
cache.m6g.8xlarge	107000	32
cache.m6g.12xlarge	160900	48
cache.m6g.16xlarge	214577	64
cache.c1.xlarge	6600	8

Tipo di nodo	max_cache_memory (in megabyte)	num_threads
cache.r3.large	13800	2
cache.r3.xlarge	29100	4
cache.r3.2xlarge	59600	8
cache.r3.4xlarge	120600	16
cache.r3.8xlarge	120600	32
cache.r4.large	12590	2
cache.r4.xlarge	25652	4
cache.r4.2xlarge	51686	8
cache.r4.4xlarge	103815	16
cache.r4.8xlarge	208144	32
cache.r4.16xlarge	416776	64
cache.r5.large	13387	2
cache.r5.xlarge	26953	4
cache.r5.2xlarge	54084	8
cache.r5.4xlarge	108347	16
cache.r5.12xlarge	325400	48
cache.r5.24xlarge	650869	96
cache.r6g.large	13387	2
cache.r6g.xlarge	26953	4
cache.r6g.2xlarge	54084	8

Tipo di nodo	max_cache_memory (in megabyte)	num_threads
cache.r6g.4xlarge	108347	16
cache.r6g.8xlarge	214577	32
cache.r6g.12xlarge	325400	48
cache.r6g.16xlarge	429154	64
cache.c7gn.large	3164	2
cache.c7gn.xlarge	6537	4
cache.c7gn.2xlarge	13248	8
cache.c7gn.4xlarge	26671	16
cache.c7gn.8xlarge	53516	32
cache.c7gn.12xlarge	325400	48
cache.c7gn.16xlarge	108347	64

Note

Tutte le istanze T2 vengono create in Amazon Virtual Private Cloud (Amazon VPC).

Scalabilità ElastiCache per Memcached

Scalabilità ElastiCache per Memcached

ElastiCache Serverless si adatta automaticamente al traffico del carico di lavoro man mano che aumenta o diminuisce. Per ogni cache ElastiCache Serverless, monitora ElastiCache continuamente l'utilizzo di risorse come CPU, memoria e rete. Quando una di queste risorse è limitata, ElastiCache Serverless esegue la scalabilità orizzontale aggiungendo un nuovo shard e ridistribuendo i dati sul nuovo shard, senza tempi di inattività dell'applicazione. È possibile monitorare le risorse utilizzate dalla cache monitorando la `BytesUsedForCache` metrica per l'archiviazione dei dati nella `CloudWatch` cache e `(E)CPU` per l'utilizzo dell'elaborazione. `ElastiCacheProcessingUnits`

Impostazione dei limiti di dimensionamento per gestire i costi

Puoi scegliere di configurare un utilizzo massimo dell'archiviazione di dati e delle `E)CPU` al secondo per controllare i costi della cache. In questo modo ti assicuri che l'utilizzo della cache non superi mai il limite massimo configurato.

Se imposti un dimensionamento massimo, l'applicazione potrebbe subire una riduzione delle prestazioni quando la cache raggiunge il limite massimo. Quando imposti un valore massimo di archiviazione dei dati nella cache e lo spazio di archiviazione dei dati della cache raggiunge il massimo, inizierà a rimuovere i dati dalla cache utilizzando la logica ElastiCache LRU. Quando imposti un valore massimo di `E)CPU`/secondo e l'utilizzo di calcolo del carico di lavoro supera questo valore, inizieranno a limitare le richieste Memcached. ElastiCache

Se imposti un limite massimo su `BytesUsedForCache` o `ElastiCacheProcessingUnits`, ti consigliamo vivamente di impostare un `CloudWatch` allarme a un valore inferiore al limite massimo in modo da ricevere una notifica quando la cache funziona in prossimità di questi limiti. Ti consigliamo di impostare l'allarme al 75% del limite massimo configurato. Consulta la documentazione su come configurare gli `CloudWatch` allarmi.

Prescalabilità con Serverless ElastiCache

ElastiCache Prescalabilità senza server

Con il preridimensionamento, chiamato anche preriscaldamento, è possibile impostare limiti minimi supportati per la cache. ElastiCache È possibile impostare questi valori minimi per le unità di ElastiCache elaborazione (`E)CPU`) al secondo o per l'archiviazione dei dati. Ciò può essere utile

in preparazione agli eventi di scalabilità previsti. Ad esempio, se una società di giochi prevede un aumento di 5 volte degli accessi entro il primo minuto dal lancio del nuovo gioco, può predisporre la cache per questo significativo picco di utilizzo.

È possibile eseguire la prescalabilità utilizzando la ElastiCache console, la CLI o l'API. ElastiCache Serverless aggiorna le ECPU disponibili al secondo nella cache entro 60 minuti e invia una notifica di evento quando viene completato l'aggiornamento del limite minimo.

Come funziona la prescalabilità

Quando il limite minimo per ECPU/secondo o l'archiviazione dei dati viene aggiornato tramite console, CLI o API, il nuovo limite è disponibile entro 1 ora. ElastiCache Serverless supporta 30.000 ECPUS/secondo su una cache vuota e fino a 90.000 ECPUS/sec quando si utilizza la funzione Read from Replica. ElastiCache può raddoppiare le ECPU al secondo ogni 10-12 minuti. Questa velocità di scalabilità è sufficiente per la maggior parte dei carichi di lavoro. Se prevedi che un evento di scalabilità imminente potrebbe superare questa frequenza, ti consigliamo di impostare le ECPU/secondo minime sul picco di ECPUS/sec previsto almeno 60 minuti prima dell'evento di picco. In caso contrario, l'applicazione potrebbe riscontrare una latenza e una limitazione delle richieste elevate.

Una volta completato l'aggiornamento del limite minimo, ElastiCache Serverless inizierà a contabilizzare il nuovo numero minimo di ECPU al secondo o il nuovo spazio di archiviazione minimo. Ciò si verifica anche se l'applicazione non esegue richieste sulla cache o se l'utilizzo dell'archiviazione dei dati è inferiore al minimo. Quando si abbassa il limite minimo rispetto all'impostazione corrente, l'aggiornamento è immediato, quindi ElastiCache Serverless inizierà immediatamente a misurare il nuovo limite minimo.

Note

- Quando imposti un limite di utilizzo minimo, ti viene addebitato il costo di tale limite anche se l'utilizzo effettivo è inferiore al limite di utilizzo minimo. All'utilizzo dell'ECPU o dell'archiviazione dei dati che supera il limite di utilizzo minimo viene addebitata la tariffa normale. Ad esempio, se imposti un limite di utilizzo minimo di 100.000 ECPU al secondo, ti verranno addebitati almeno 1,224 USD all'ora (utilizzando i prezzi delle ECPU in us-east-1), anche se l'utilizzo è inferiore al minimo stabilito.
- ElastiCache Serverless supporta la scala minima richiesta a livello aggregato sulla cache. ElastiCache Serverless supporta anche un massimo di 30.000 ECPU/secondo per slot (90.000 ECPU/secondo quando si utilizza Read from Replica utilizzando connessioni

READONLY). Come best practice, l'applicazione dovrebbe garantire che la distribuzione delle chiavi tra gli slot Redis e il traffico tra le chiavi siano il più uniformi possibile.

Impostazione dei limiti di scalabilità utilizzando la console e AWS CLI

Impostazione dei limiti di scalabilità tramite la console AWS

1. Accedi AWS Management Console e apri la ElastiCache console all'[indirizzo https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/).
2. Nel riquadro di navigazione scegli il motore in esecuzione sulla cache da modificare.
3. Viene visualizzato un elenco di cache che eseguono il motore scelto.
4. Scegli la cache da modificare facendo clic sul pulsante di opzione a sinistra del nome della cache.
5. Scegliere Actions (Operazioni), quindi Modify (Modifica).
6. In Limiti di utilizzo, imposta i limiti di memoria o elaborazione appropriati.
7. Fai clic su Visualizza l'anteprima delle modifiche e quindi su Salva le modifiche.

Impostazione dei limiti di scalabilità utilizzando il AWS CLI

Per modificare i limiti di scalabilità utilizzando la CLI, utilizza l' `modify-serverless-cache` API.

Linux:

```
aws elasticache modify-serverless-cache --serverless-cache-name <cache name> \  
--cache-usage-limits 'DataStorage={Minimum=10,Maximum=100,Unit=GB},  
ECPUPerSecond={Minimum=1000,Maximum=100000}'
```

Windows:

```
aws elasticache modify-serverless-cache --serverless-cache-name <cache name> ^  
--cache-usage-limits 'DataStorage={Minimum=10,Maximum=100,Unit=GB},  
ECPUPerSecond={Minimum=1000,Maximum=100000}'
```

Rimozione dei limiti di dimensionamento tramite la CLI

Per rimuovere i limiti di scalabilità utilizzando la CLI, imposta il parametro Limite minimo e massimo su 0.

Linux:

```
aws elasticache modify-serverless-cache --serverless-cache-name <cache name> \  
--cache-usage-limits 'DataStorage={Minimum=0,Maximum=0,Unit=GB},  
ECPUPerSecond={Minimum=0,Maximum=0}'
```

Windows:

```
aws elasticache modify-serverless-cache --serverless-cache-name <cache name> ^  
--cache-usage-limits 'DataStorage={Minimum=0,Maximum=0,Unit=GB},  
ECPUPerSecond={Minimum=0,Maximum=0}'
```

Scalabilità ElastiCache per cluster progettati autonomamente da Memcached

Raramente la quantità dei dati che un'applicazione deve elaborare è statica. Aumenta e diminuisce secondo le fluttuazioni della domanda del tuo business. Se gestisci da solo la tua cache, è necessario disporre dell'hardware sufficiente per soddisfare i picchi della domanda e questo può comportare costi cospicui. Con Amazon ElastiCache puoi scalare per soddisfare la domanda attuale, pagando solo per ciò che usi. ElastiCache ti consente di scalare la cache in base alla domanda.

I contenuti seguenti ti danno la possibilità di trovare l'argomento adatto alle operazioni di scalabilità di tuo interesse.

Dimensionamento dei cluster Memcached

Azione	Argomento
Dimensionamento orizzontale	Aggiunta di nodi a un cluster
Dimensionamento verticale	Eliminazione di nodi da un cluster
Modifica dei tipi di nodo	Dimensionamento verticale di Memcached

I cluster Memcached sono composti da 1 a 60 nodi. Il dimensionamento orizzontale e verticale di un cluster Memcached è semplice come l'aggiunta al o la rimozione di nodi dal cluster.

[Se hai bisogno di più di 60 nodi in un cluster Memcached o più di 300 nodi in totale in una AWS regione, compila il modulo di richiesta di aumento del ElastiCache limite all'indirizzo https://aws.amazon.com/contact-us//.elasticache-node-limit-request](https://aws.amazon.com/contact-us//.elasticache-node-limit-request)

Poiché è possibile ripartire i dati su tutti i nodi in un cluster Memcached, il dimensionamento verso l'alto a un tipo di nodo con una memoria di dimensioni maggiori è richiesto raramente. Tuttavia, poiché il motore Memcached non rende i dati permanenti, se si effettua il dimensionamento a un tipo di nodo differente, il nuovo cluster è inizialmente vuoto a meno che non venga popolato dall'applicazione.

Argomenti

- [Dimensionamento orizzontale dei cluster Memcached](#)
- [Dimensionamento verticale di Memcached](#)

Dimensionamento orizzontale dei cluster Memcached

Il motore Memcached supporta il partizionamento dei dati su più nodi. Per questo motivo, il dimensionamento orizzontale dei cluster Memcached è più semplice. Un cluster Memcached può avere da 1 a 60 nodi. Per dimensionare orizzontalmente un cluster Memcached è sufficiente aggiungere o rimuovere i nodi.

[Se hai bisogno di più di 60 nodi in un cluster Memcached o più di 300 nodi in totale in una AWS regione, compila il modulo di richiesta di aumento del ElastiCache limite all'indirizzo <https://aws.amazon.com/contact-us//.elasticache-node-limit-request>](#)

Gli argomenti che seguono illustrano nel dettaglio come dimensionare un cluster Memcached verso l'alto o verso il basso aggiungendo o rimuovendo nodi.

- [Aggiunta di nodi a un cluster](#)
- [Eliminazione di nodi da un cluster](#)

Ogni volta che si cambia il numero di nodi di un cluster Memcached, è necessario rimappare almeno una parte del keyspace in modo da mapparlo sul nodo corretto. Per informazioni più dettagliate sul load balancer del cluster Memcached, consulta [Configurazione del client ElastiCache per un efficiente bilanciamento del carico](#):

Se utilizzi il rilevamento automatico sul cluster Memcached, non è necessario modificare gli endpoint nell'applicazione quando aggiungi o rimuovi nodi. Per ulteriori informazioni sull'individuazione automatica, consulta [Identificazione automatica dei nodi nel cluster](#). Se non utilizzi l'individuazione automatica, ogni volta che modifichi il numero di nodi nel cluster Memcached è necessario aggiornare gli endpoint nell'applicazione.

Dimensionamento verticale di Memcached

Quando effettui il dimensionamento verso l'alto o verso il basso di un cluster Memcached, devi creare un nuovo cluster. Al momento della creazione, i cluster Memcached sono sempre vuoti, a meno che l'applicazione non li popoli.

Important

Se esegui il dimensionamento a un tipo di nodo più piccolo, assicurati che il tipo di nodo più piccolo sia adeguato ai dati e all'overhead. Per ulteriori informazioni, consulta [Select cache node size](#).

Argomenti

- [Dimensionamento verticale di Memcached \(console\)](#)
- [Dimensionamento verticale di Memcached \(AWS CLI\)](#)
- [Scalabilità verticale di Memcached \(API\) ElastiCache](#)

Dimensionamento verticale di Memcached (console)

La procedura seguente illustra come scalare verticalmente il cluster utilizzando la console ElastiCache

Per dimensionare verticalmente un cluster Memcached (console)

1. Creare un nuovo cluster con il nuovo tipo di nodo. Per ulteriori informazioni, consulta [Creazione di un cluster Memcached \(console\)](#).
2. Nell'applicazione, aggiornare gli endpoint con gli endpoint del nuovo cluster. Per ulteriori informazioni, consulta [Individuazione degli endpoint di un cluster \(console\)](#).
3. Eliminare il cluster precedente. Per ulteriori informazioni, consulta [Deleting a new node in Memcached](#).

Dimensionamento verticale di Memcached (AWS CLI)

La procedura seguente descrive le diverse fasi per eseguire il dimensionamento verticale del cluster di cache Memcached tramite AWS CLI.

Per dimensionare verticalmente un cluster di cache Memcached (AWS CLI)

1. Creare un nuovo cluster di cache con il nuovo tipo di nodo. Per ulteriori informazioni, consulta [Creating clusters with the CLI](#).
2. Nell'applicazione, aggiornare gli endpoint con gli endpoint del nuovo cluster. Per ulteriori informazioni, consulta [Individuazione degli endpoint \(AWS CLI\)](#).
3. Eliminare il cluster di cache precedente. Per ulteriori informazioni, consulta [Utilizzo di AWS CLI](#).

Scalabilità verticale di Memcached (API) ElastiCache

La procedura seguente illustra come scalare verticalmente il cluster di cache Memcached utilizzando l'API. ElastiCache

Per scalare verticalmente un cluster di cache Memcached (API) ElastiCache

1. Creare un nuovo cluster di cache con il nuovo tipo di nodo. Per ulteriori informazioni, consultare [Creazione di un cluster \(ElastiCache API\)](#)
2. Nell'applicazione, aggiornare gli endpoint sugli endpoint del nuovo cluster di cache. Per ulteriori informazioni, consulta [Ricerca degli endpoint \(API ElastiCache\)](#).
3. Eliminare il cluster di cache precedente. Per ulteriori informazioni, consulta [Utilizzo dell'API ElastiCache](#).

Assegnazione di tag delle risorse ElastiCache

Per semplificare la gestione delle risorse ElastiCache, è possibile assegnare metadati personalizzati a ciascuna risorsa sotto forma di tag. I tag consentono di categorizzare le tue risorse AWS in modi diversi, ad esempio, per scopo, proprietario o ambiente. Questa caratteristica è molto utile quando hai tante risorse dello stesso tipo in quanto puoi rapidamente individuare una risorsa specifica in base ai tag assegnati. Questo argomento descrive i tag e mostra come crearli.

Warning

Come da best practice, è consigliabile non includere dati sensibili nei tag.

Nozioni di base sui tag

Un tag è un'etichetta che assegni a una risorsa AWS. Ogni tag è composto da una chiave e da un valore opzionale, entrambi personalizzabili. I tag consentono di categorizzare le tue risorse AWS in modi diversi, ad esempio, per scopo o proprietario. Ad esempio, puoi definire un set di tag per i cluster del tuo account ElastiCache e monitorare così ogni proprietario dell'istanza e il gruppo di utenti.

Ti consigliamo di creare un set di chiavi di tag in grado di soddisfare i requisiti di ciascun tipo di risorsa. Con un set di chiavi di tag coerente, la gestione delle risorse risulta semplificata. Puoi cercare e filtrare le risorse in base ai tag aggiunti. Per ulteriori informazioni sulle modalità di implementazione di una strategia efficace di applicazione di tag alle risorse, consulta il [whitepaper AWS Best practice per l'applicazione di tag](#).

I tag non hanno alcun significato semantico per ElastiCache e vengono interpretati rigorosamente come una stringa di caratteri. Inoltre, i tag non vengono assegnati automaticamente alle risorse. Puoi modificare chiavi e valori di tag e rimuovere tag da una risorsa in qualsiasi momento. È possibile impostare il valore di un tag su null. Se aggiungi un tag con la stessa chiave di un tag esistente a una risorsa specifica, il nuovo valore sovrascrive quello precedente. Se elimini una risorsa, verranno eliminati anche tutti i tag associati alla risorsa. Inoltre, se si aggiungono o eliminano tag in un gruppo di replica, verranno aggiunti o rimossi anche i tag di tutti i nodi del gruppo di replica.

Puoi lavorare con i tag utilizzando la AWS Management Console, lo AWS CLI e l'API ElastiCache.

Se utilizzi (IAM), puoi controllare quali utenti nel tuo account AWS dispongono dell'autorizzazione per creare, modificare o eliminare i tag. Per ulteriori informazioni, consulta [Autorizzazioni a livello di risorsa](#).

Risorse cui è possibile associare tag

Puoi assegnare tag alla maggior parte delle risorse ElastiCache già esistenti nel tuo account. Nella tabella seguente sono elencate le risorse che supportano il tagging. Se usi il comando AWS Management Console, puoi applicare tag a risorse utilizzando l'[Editor di tag](#). Alcune schermate relative alle risorse ti permettono di specificare i tag per una risorsa quando crei la risorsa, ad esempio un tag con la chiave con nome e un valore specificato. Nella maggior parte dei casi, la console applica i tag subito dopo la creazione della risorsa, anziché durante il processo di creazione. La console può organizzare le risorse in base al relativo tag Nome ma questo tag non ha un significato semantico per il servizio ElastiCache

Inoltre, alcune operazioni per la creazione di risorse ti consentono di specificare tag per una risorsa durante la sua creazione. Se i tag non possono essere applicati durante la creazione della risorsa, eseguiamo il rollback del processo di creazione della risorsa. Ciò fa sì che le risorse vengano create con i tag oppure che non vengano create affatto, nonché che nessuna risorsa sia mai sprovvista di tag. Il tagging delle risorse in fase di creazione ti permette di evitare di eseguire script di tagging personalizzati dopo la creazione delle risorse.

Se utilizzi l'API Amazon ElastiCache la CLI AWS o un SDK AWS, puoi usare il parametro Tags sull'operazione API ElastiCache pertinente per applicare i tag. Questi sono:

- `CreateServerlessCache`
- `CreateCacheCluster`
- `CreateCacheParameterGroup`
- `CreateCacheSecurityGroup`
- `CreateCacheSubnetGroup`
- `PurchaseReservedCacheNodesOffering`

Nella seguente tabella sono descritte le risorse ElastiCache a cui puoi associare i tag, nonché le risorse che possono essere associate a tag utilizzando l'API ElastiCache, la CLI AWS o un SDK AWS.

Support dell'assegnazione di tag per le risorse ElastiCache

support dei tag	Supporta l'applicazione di tag in fase di creazione
Sì	Sì
Sì	Sì
Sì	Sì
Sì	Sì
Sì	Sì
Sì	Sì

Puoi applicare autorizzazioni basate su tag a livello di risorsa nelle policy IAM alle operazioni dell'API ElastiCache che supportano il l'assegnazione di tag in fase di creazione per implementare un controllo granulare sugli utenti e sui gruppi che associano tag alle risorse in fase di creazione. Le risorse vengono adeguatamente protette dalla creazione, ovvero tag che vengono applicati immediatamente alle risorse. Pertanto qualsiasi autorizzazione basata su tag a livello di risorsa che controlla l'uso delle risorse risulta immediatamente valida. Le risorse possono essere monitorate e segnalate con maggiore precisione. Puoi applicare l'uso del tagging alle nuove risorse e controllare quali chiavi e valori di tag sono impostati per le risorse.

Per ulteriori informazioni, consulta [Esempio: assegnazione di tag alle risorse](#).

Per ulteriori informazioni sul tagging delle risorse per la fatturazione, vedere [Monitoraggio dei costi con i tag di allocazione dei costi](#).

Limitazioni applicate ai tag

Si applicano le seguenti limitazioni di base ai tag:

- Numero massimo di tag per risorsa: 50
- Per ciascuna risorsa, ogni chiave del tag deve essere univoca e ogni chiave del tag può avere un solo valore.
- La lunghezza massima della chiave è 128 caratteri Unicode in formato UTF-8.
- La lunghezza massima del valore è 256 caratteri Unicode in formato UTF-8.
- Sebbene ElastiCache consenta l'uso di qualsiasi carattere nei tag, altri servizi presentano limitazioni più restrittive. I caratteri consentiti nei servizi sono: lettere, numeri e spazi rappresentabili in formato UTF-8 e i seguenti caratteri speciali + - = . _ : / @.
- Per chiavi e valori di tag viene fatta la distinzione tra maiuscole e minuscole.
- Il prefisso `aws :` è riservato per l'uso di AWS. Se il tag ha una chiave di tag con questo prefisso, non puoi modificare o eliminare la chiave o il valore de tag. I tag con il prefisso `aws :` non vengono conteggiati per il limite del numero di tag per risorsa.

Non puoi interrompere, arrestare o eliminare una risorsa solo sulla base dei relativi tag. Devi specificare il relativo identificatore. Ad esempio, per eliminare gli snapshot associato a una chiave di tag denominata `DeleteMe`, devi utilizzare l'operazione `DeleteSnapshot` con gli identificatori di risorsa degli snapshot, ad esempio `snap-1234567890abcdef0`.

Per ulteriori informazioni sulle risorse ElastiCache che è possibile taggare, vedere [Risorse cui è possibile associare tag](#).

Esempio: assegnazione di tag alle risorse

- Creazione di una cache serverless utilizzando i tag

```
aws elasticache create-serverless-cache \  
  --serverless-cache-name CacheName \  
  --engine memcached \  
  --tags Key="Cost Center", Value="1110001" Key="project",Value="XYZ"
```

- Aggiunta di tag a una cache serverless

```
aws elasticache add-tags-to-resource \  
  --resource-name arn:aws:elasticache:us-east-1:111111222233:serverlesscache:my-cache \  
  --tags Key="Cost Center", Value="1110001" Key="project",Value="XYZ"
```

```
--tags Key="project",Value="XYZ" Key="Elasticache",Value="Service"
```

- Creazione di un cluster di cache con i tag.

```
aws elasticache create-cache-cluster \  
--cluster-id testing-tags \  
--cluster-description cluster-test \  
--cache-subnet-group-name test \  
--cache-node-type cache.t2.micro \  
--engine memcached \  
--tags Key="project",Value="XYZ" Key="Elasticache",Value="Service"
```

Esempi di policy di controllo degli accessi basati sui tag

1. Consentire l'operazione `AddTagsToResource` a un cluster solo se il cluster ha il tag `Project=XYZ`.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": "elasticache:AddTagsToResource",  
      "Resource": [  
        "arn:aws:elasticache:*:*:cluster:*"  
      ],  
      "Condition": {  
        "StringEquals": {  
          "aws:ResourceTag/Project": "XYZ"  
        }  
      }  
    }  
  ]  
}
```

2. Consentire l'operazione `RemoveTagsFromResource` da un gruppo di replica se contiene i tag progetto e servizio e le chiavi sono diverse da `Project` e `Service`.

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

```

```

    {
      "Effect": "Allow",
      "Action": "elasticache:RemoveTagsFromResource",
      "Resource": [
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Service": "Elasticache",
          "aws:ResourceTag/Project": "XYZ"
        },
        "ForAnyValue:StringNotEqualsIgnoreCase": {
          "aws:TagKeys": [
            "Project",
            "Service"
          ]
        }
      }
    }
  ]
}

```

3. Consentire `AddTagsToResource` a qualsiasi risorsa solo se i tag sono diversi da progetto e servizio.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "elasticache:AddTagsToResource",
      "Resource": [
        "arn:aws:elasticache:*:*:*:*"
      ],
      "Condition": {
        "ForAnyValue:StringNotEqualsIgnoreCase": {
          "aws:TagKeys": [
            "Service",
            "Project"
          ]
        }
      }
    }
  ]
}

```



```
]
}
```

4. Rifiuto dell'azione `CreateCacheCluster` se il tag della richiesta `Project` manca o è diverso da `Dev`, `QA` o `Prod`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*",
        "arn:aws:elasticache:*:*:securitygroup:*",
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ]
    },
    {
      "Effect": "Deny",
      "Action": [
        "elasticache:CreateCacheCluster"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*"
      ],
      "Condition": {
        "Null": {
          "aws:RequestTag/Project": "true"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster",
        "elasticache:AddTagsToResource"
      ],
      "Resource": "arn:aws:elasticache:*:*:cluster:*",
      "Condition": {
```

```
    "StringEquals": {
      "aws:RequestTag/Project": [
        "Dev",
        "Prod",
        "QA"
      ]
    }
  }
}
```

Per informazioni correlate sulle chiavi di condizione, consulta [Utilizzo delle chiavi di condizione](#).

Monitoraggio dei costi con i tag di allocazione dei costi

Quando aggiungi i tag di allocazione dei costi alle tue risorse in Amazon ElastiCache, puoi tenere traccia dei costi raggruppando le spese nelle fatture mediante i valori dei tag della risorsa.

Un tag di allocazione dei costi ElastiCache è una coppia chiave-valore definita e associata a una risorsa ElastiCache. La chiave e il valore fanno distinzione tra maiuscole e minuscole. Puoi utilizzare una chiave di tag per definire una categoria e il valore come una voce di tale categoria. Ad esempio, puoi definire una chiave di tag `CostCenter` e un valore di tag `10010`, a indicare che la risorsa è assegnata al centro di costo 10010. È anche possibile usare i tag per indicare le risorse come risorse utilizzate a scopo di test o produzione tramite una chiave, ad esempio `Environment`, e tramite valori, ad esempio `test` o `production`. È consigliabile utilizzare un set coerente di chiavi di tag per agevolare il monitoraggio dei costi associati alle risorse.

Per organizzare le fatture AWS al fine di riflettere la struttura dei costi, puoi utilizzare i tag di allocazione dei costi. Per eseguire questa operazione, registrati per far sì che la fattura del tuo account AWS includa i valori di chiave di tag. Per visualizzare il costo delle risorse combinate, puoi organizzare le informazioni di fatturazione in base alle risorse con gli stessi valori di chiave di tag. Puoi ad esempio applicare tag a numerose risorse con un nome di applicazione specifico, quindi organizzare le informazioni di fatturazione per visualizzare il costo totale dell'applicazione in più servizi.

Puoi anche combinare i tag per monitorare i costi con un livello di dettagli maggiore. Ad esempio, per monitorare i costi di servizio per regione, puoi utilizzare le chiavi di tag `Service` e `Region`. Su una risorsa potresti avere i valori `ElastiCache` e `Asia Pacific (Singapore)`, mentre su un'altra

risorsa potresti avere i valori ElastiCache e Europe (Frankfurt). Puoi quindi vedere i tuoi costi totali ElastiCache suddivisi tra le regioni. Per ulteriori informazioni, consulta la pagina sull'[utilizzo dei tag per l'allocazione dei costi](#) nella Guida per l'utente di AWS Billing.

Puoi anche aggiungere i tag di allocazione dei costi ElastiCache ai cluster Memcached. Quando aggiungi, elenchi, modifichi, copi o rimuovi un tag, l'operazione viene applicata solo al cluster specificato.

Caratteristiche dei tag di allocazione dei costi di ElastiCache

- I tag di allocazione dei costi vengono applicati alle risorse ElastiCache specificate nelle operazioni API e CLI come ARN. Il tipo di risorsa sarà un cluster.

ARN di esempio: `arn:aws:elasticache:<region>:<customer-id>:<resource-type>:<resource-name>`

Memcached: i tag vengono applicati solo ai cluster.

Arn di esempio: `arn:aws:elasticache:us-west-2:1234567890:cluster:my-cluster`

- La chiave di tag corrisponde al nome obbligatorio del tag. Il valore della stringa della chiave può essere composto da 1 a 128 caratteri Unicode e non può avere il prefisso `aws:`. La stringa può contenere solo il set di lettere, numeri, spazi vuoti, caratteri di sottolineatura (`_`), punti (`.`), virgole (`:`), barre rovesciate (`\`), segni di uguale (`=`), più (`+`), trattini (`-`) o chioccioline (`@`).
- Un valore tag è il valore opzionale del tag. Il valore di stringa del valore può essere composto da 1 a 256 caratteri Unicode e non può avere il prefisso `aws:`. La stringa può contenere solo il set di lettere, numeri, spazi vuoti, caratteri di sottolineatura (`_`), punti (`.`), virgole (`:`), barre rovesciate (`\`), segni di uguale (`=`), più (`+`), trattini (`-`) o chioccioline (`@`).
- Una risorsa ElastiCache può avere un massimo di 50 tag.
- I valori non devono essere necessariamente univoci in un set di tag. Ad esempio, puoi avere un set di tag dove le chiavi `Service` e `Application` hanno entrambe il valore `ElastiCache`.

AWS non applica significati semantici ai tag. I tag vengono interpretati rigorosamente come stringhe di caratteri. AWS non imposta automaticamente i tag sulle risorse ElastiCache.

Gestione dei tag di allocazione dei costi tramite la console AWS CLI

Puoi utilizzare l'AWS CLI per aggiungere, modificare o rimuovere i tag di allocazione dei costi.

I tag di allocazione dei costi vengono applicati a ElastiCache per i cluster Memcached. Il cluster a cui aggiungere tag viene specificato mediante un Amazon Resource Name (ARN).

Arn di esempio `arn:aws:elasticache:us-west-2:1234567890:cluster:my-cluster`

Arn di esempio `arn:aws:elasticache:us-west-2:1234567890:cluster:my-cluster`

Argomenti

- [Elenco dei tag utilizzando la AWS CLI](#)
- [Aggiunta di tag utilizzando la AWS CLI](#)
- [Modifica dei tag utilizzando l' AWS CLI](#)
- [Rimozione dei tag utilizzando l' AWS CLI](#)

Elenco dei tag utilizzando la AWS CLI

È possibile utilizzare AWS CLI per elencare i tag su una risorsa esistente ElastiCache tramite l'operazione [list-tags-for-resource](#).

Il codice seguente utilizza l'AWS CLI per elencare i tag sul cluster Memcached `my-cluster` nella regione `us-west-2`.

Per Linux, macOS o Unix:

```
aws elasticache list-tags-for-resource \  
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster
```

Per Windows:

```
aws elasticache list-tags-for-resource ^  
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster
```

L'output di questa operazione sarà simile al seguente, una lista di tutti i tag sulla risorsa.

```
{  
  "TagList": [  
    {
```

```

    "Value": "10110",
    "Key": "CostCenter"
  },
  {
    "Value": "EC2",
    "Key": "Service"
  }
]
}

```

Se non sono presenti tag nella risposta, l'operazione restituisce una mappa TagList vuota.

```

{
  "TagList": []
}

```

Per ulteriori informazioni sulle proprietà di ElastiCache [list-tags-for-resource](#), consultare AWS CLI.

Aggiunta di tag utilizzando la AWS CLI

È possibile utilizzare la AWS CLI per aggiungere i tag a una risorsa ElastiCache esistente tramite l'operazione CLI [add-tags-to-resource](#). Se la nuova chiave di tag non esiste sulla risorsa, la chiave e il valore vengono aggiunti alla risorsa. Se la chiave esiste già sulla risorsa, il valore associato a quella chiave viene aggiornato al nuovo valore.

Il codice seguente utilizza la AWS CLI per aggiungere le chiavi Service e Region con i valori elasticache e us-west-2 rispettivamente sul cluster my-cluster nella regione us-west-2.

Per Linux, macOS o Unix:

```

aws elasticache add-tags-to-resource \
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster \
  --tags Key=Service,Value=elasticache \
    Key=Region,Value=us-west-2

```

Per Windows:

```

aws elasticache add-tags-to-resource ^
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster ^
  --tags Key=Service,Value=elasticache ^
    Key=Region,Value=us-west-2

```

L'output di questa operazione sarà simile al seguente, una lista di tutti i tag sulla risorsa in base all'operazione.

```
{
  "TagList": [
    {
      "Value": "elasticache",
      "Key": "Service"
    },
    {
      "Value": "us-west-2",
      "Key": "Region"
    }
  ]
}
```

Per ulteriori informazioni sulle proprietà di ElastiCache [add-tags-to-resource](#), consultare AWS CLI.

Puoi anche utilizzare l'AWS CLI per aggiungere i tag a un cluster quando crei un nuovo cluster mediante l'operazione [create-cache-cluster](#). Non è possibile aggiungere tag durante la creazione di un cluster mediante la console di gestione ElastiCache. Dopo aver creato il cluster, puoi quindi utilizzare la console per aggiungere i tag al cluster.

Modifica dei tag utilizzando l' AWS CLI

Puoi utilizzare il plugin AWS CLI Per modificare i tag su un cluster ElastiCache per cluster Memcached.

Per modificare i tag:

- Utilizzare [add-tags-to-resource](#) per aggiungere un nuovo tag e valore o per modificare il valore associato a un tag esistente.
- Utilizzare [remove-tags-from-resource](#) per rimuovere i tag specificati dalla risorsa.

L'output da entrambe le operazioni sarà un elenco di tag e i relativi valori sul cluster specificato.

Rimozione dei tag utilizzando l' AWS CLI

È possibile utilizzare la AWS CLI per rimuovere i tag da un cluster ElastiCache for Memcached esistente tramite l'operazione [remove-tags-from-resource](#).

Il codice seguente utilizza la AWS CLI per rimuovere i tag con le chiavi Service e Region cluster `my-cluster` nella regione `us-west-2`.

Per Linux, macOS o Unix:

```
aws elasticache remove-tags-from-resource \  
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster \  
  --tag-keys PM Service
```

Per Windows:

```
aws elasticache remove-tags-from-resource ^  
  --resource-name arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster ^  
  --tag-keys PM Service
```

L'output di questa operazione sarà simile al seguente, una lista di tutti i tag sulla risorsa in base all'operazione.

```
{  
  "TagList": []  
}
```

Per ulteriori informazioni sulle proprietà di ElastiCache [remove-tags-from-resource](#), consultare AWS CLI.

Gestione dei tag di allocazione dei costi tramite l'API ElastiCache

Puoi utilizzare l'API ElastiCache per aggiungere, modificare o rimuovere i tag di allocazione dei costi.

I tag di allocazione dei costi vengono applicati ai cluster ElastiCache for Memcached. Il cluster a cui aggiungere tag viene specificato mediante un Amazon Resource Name (ARN).

Arn di esempio `arn:aws:elasticache:us-west-2:1234567890:cluster:my-cluster`

Argomenti

- [Elenco dei tag con l'API ElastiCache](#)
- [Aggiunta di tag con l'API ElastiCache](#)
- [Modifica dei tag utilizzando l'API ElastiCache](#)
- [Rimozione dei tag utilizzando l'API ElastiCache](#)

Elenco dei tag con l'API ElastiCache

È possibile utilizzare l'API ElastiCache per elencare i tag su una risorsa esistente tramite l'operazione [ListTagsForResource](#).

Il codice seguente utilizza l'API ElastiCache per elencare i tag sulla risorsa nella regione us-west-2 my-cluster.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=ListTagsForResource  
&ResourceName=arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Version=2015-02-02  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Aggiunta di tag con l'API ElastiCache

È possibile utilizzare l'API ElastiCache per aggiungere i tag a un cluster ElastiCache esistente tramite l'operazione [AddTagsToResource](#). Se la nuova chiave di tag non esiste sulla risorsa, la chiave e il valore vengono aggiunti alla risorsa. Se la chiave esiste già sulla risorsa, il valore associato a quella chiave viene aggiornato al nuovo valore.

Il codice seguente utilizza l'API ElastiCache per aggiungere le chiavi Service e Region con i valori elasticache e us-west-2 rispettivamente per le risorse my-cluster nella regione us-west-2.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=AddTagsToResource  
&ResourceName=arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster  
&SignatureVersion=4  
&SignatureMethod=HmacSHA256  
&Tags.member.1.Key=Service  
&Tags.member.1.Value=elasticache  
&Tags.member.2.Key=Region  
&Tags.member.2.Value=us-west-2  
&Version=2015-02-02  
&Timestamp=20150202T192317Z  
&X-Amz-Credential=<credential>
```

Per ulteriori informazioni, consulta [AddTagsToResource](#) nella Guida di riferimento delle API di Amazon ElastiCache.

Modifica dei tag utilizzando l'API ElastiCache

Puoi utilizzare l'API ElastiCache per modificare i tag su un cluster ElastiCache.

Per modificare il valore di un tag:

- Utilizzare l'operazione [AddTagsToResource](#) per aggiungere un nuovo tag e valore o per modificare il valore associato a un tag esistente.
- Utilizzare [RemoveTagsFromResource](#) per rimuovere i tag dalla risorsa.

L'output da entrambe le operazioni sarà un elenco di tag e dei relativi valori sulla risorsa specificata.

Utilizzare [RemoveTagsFromResource](#) per rimuovere i tag dalla risorsa.

Rimozione dei tag utilizzando l'API ElastiCache

È possibile utilizzare l'API ElastiCache per rimuovere i tag da un cluster ElastiCache for Memcached esistente tramite l'operazione [RemoveTagsFromResource](#).

Il codice seguente utilizza l'API ElastiCache per rimuovere i tag con le chiavi Service e Region dal cluster `my-cluster` nella regione `us-west-2`.

```
https://elasticache.us-west-2.amazonaws.com/  
  ?Action=RemoveTagsFromResource  
  &ResourceName=arn:aws:elasticache:us-west-2:0123456789:cluster:my-cluster  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &TagKeys.member.1=Service  
  &TagKeys.member.2=Region  
  &Version=2015-02-02  
  &Timestamp=20150202T192317Z  
  &X-Amz-Credential=<credential>
```

Utilizzo di Amazon ElastiCache Well-Architected Lens

Questa sezione descrive Amazon ElastiCache Well-Architected Lens, una raccolta di principi e linee guida per la progettazione di carichi di lavoro ElastiCache correttamente strutturati.

- ElastiCache Lens è in aggiunta a [Framework AWS Well-Architected](#).

- Ogni pilastro contiene una serie di domande per aiutare ad avviare la discussione su un'architettura ElastiCache.
- Ogni domanda contiene una serie di procedure principali insieme ai relativi punteggi per la segnalazione.
 - **Obbligatorio:** la procedura obbligatoria prima di passare in produzione (se non viene eseguita comporta un alto rischio)
 - **Best practice:** la migliore procedura per un cliente
 - **Consigliato:** la procedura consigliata ai clienti (se non viene eseguita comporta un rischio medio)
- Terminologia di Well-Architected
 - **Componente:** codice, configurazione e risorse AWS che insieme soddisfano un requisito. I componenti interagiscono con altri componenti e spesso equivalgono a un servizio nelle architetture di microservizi.
 - **Carico di lavoro:** un insieme di componenti che forniscono valore aziendale. Esempi di carichi di lavoro sono siti Web di marketing, siti Web di e-commerce, il back-end di un'app per dispositivi mobili, piattaforme di analisi e così via.

Argomenti

- [Pilastro dell'eccellenza operativa di Amazon ElastiCache Well-Architected Lens](#)
- [Pilastro della sicurezza di Amazon ElastiCache Well-Architected Lens](#)
- [Pilastro dell'affidabilità di Amazon ElastiCache Well-Architected Lens](#)
- [Pilastro dell'efficienza delle prestazioni di Amazon ElastiCache Well-Architected Lens](#)
- [Pilastro dell'ottimizzazione dei costi di Amazon ElastiCache Well-Architected Lens](#)

Pilastro dell'eccellenza operativa di Amazon ElastiCache Well-Architected Lens

Il pilastro dell'eccellenza operativa si concentra sull'esecuzione e sul monitoraggio dei sistemi per fornire valore aziendale e migliorare continuamente processi e procedure. Gli argomenti chiave includono l'automazione delle modifiche, la risposta agli eventi e la definizione degli standard per gestire le operazioni quotidiane.

Argomenti

- [EO 1: come si fa a comprendere e rispondere agli avvisi e agli eventi generati dal cluster ElastiCache?](#)
- [EO 2: quando e come si dimensionano i cluster ElastiCache esistenti?](#)
- [EO 3: come si gestiscono le risorse del cluster ElastiCache e si mantiene aggiornato il cluster?](#)
- [EO 4: come si gestiscono le connessioni dei client ai cluster ElastiCache?](#)
- [EO 5: come si distribuiscono i componenti ElastiCache per un carico di lavoro?](#)
- [EO 6: come si pianificano e si contengono gli errori?](#)
- [EO 7: come si risolvono gli eventi del motore Redis?](#)

EO 1: come si fa a comprendere e rispondere agli avvisi e agli eventi generati dal cluster ElastiCache?

Introduzione della domanda: quando si utilizzano i cluster ElastiCache, è possibile ricevere notifiche e avvisi opzionali quando si verificano eventi specifici. ElastiCache, per impostazione predefinita, registra [gli eventi](#) relativi alle risorse, come il failover, la sostituzione dei nodi, le operazioni di dimensionamento, la manutenzione programmata e altro ancora. Ogni evento include la data e l'ora, il nome e il tipo di origine e una descrizione.

Vantaggio della domanda: la capacità di comprendere e gestire i motivi alla base degli eventi che generano gli avvisi del cluster consente di operare in modo più efficace e di rispondere agli eventi in modo appropriato.

- [Obbligatorio] Rivedi gli eventi generati da ElastiCache sulla console ElastiCache (dopo aver selezionato la tua regione) o utilizzando il comando [describe-events](#) dell'[Interfaccia della linea di comando Amazon](#) (CLI AWS) e l'[API di ElastiCache](#). Configura ElastiCache in modo tale che invii notifiche sugli eventi importanti del cluster con Amazon Simple Notification Service (Amazon SNS). L'utilizzo di Amazon SNS con i cluster ti consente di intraprendere azioni programmatiche sugli eventi ElastiCache.
- Esistono due grandi categorie di eventi: eventi attuali e programmati. L'elenco degli eventi correnti include: creazione ed eliminazione delle risorse, operazioni di dimensionamento, failover, riavvio del nodo, creazione di snapshot, modifica dei parametri del cluster, rinnovo del certificato CA, eventi di errore (errore di provisioning del cluster per VPC o ENI, errori di dimensionamento per ENI ed errori di snapshot). L'elenco degli eventi pianificati include: nodo programmato per la sostituzione durante la finestra di manutenzione e sostituzione del nodo riprogrammata.

- Sebbene non sia necessario reagire immediatamente ad alcuni di questi eventi, è fondamentale esaminare tutti gli eventi di errore:
 - ElastiCache:AddCacheNodeFailed
 - ElastiCache:CacheClusterProvisioningFailed
 - Elasticache: cacheclusterScalingFailed
 - ElastiCache:cachenodesRebooted
 - ElastiCache: SnapshotFailed (solo Redis)
- [Risorse]:
 - [Gestione ElastiCache Amazon SNS](#)
 - [Notifiche di eventi Amazon SNS](#)
- [Best practice] Per automatizzare le risposte agli eventi, utilizza le funzionalità di prodotti e servizi AWS come SNS e funzioni Lambda. Segui le best practice apportando modifiche piccole, frequenti e reversibili, come codice per migliorare le tue operazioni nel tempo. Utilizza le metriche di Amazon CloudWatch per monitorare i cluster.

[Risorse]: [monitora gli endpoint di replica di lettura di Amazon ElastiCache per Redis \(modalità cluster disabilitata\) con AWS Lambda, Amazon Route 53 e Amazon SNS](#) per un caso d'uso che utilizza Lambda e SNS.

EO 2: quando e come si dimensionano i cluster ElastiCache esistenti?

Introduzione della domanda: il corretto dimensionamento del cluster ElastiCache è un atto di bilanciamento che deve essere valutato ogni volta che vengono apportate modifiche ai tipi di carico di lavoro sottostanti. Il tuo obiettivo è operare con l'ambiente delle dimensioni giuste per il tuo carico di lavoro.

Vantaggio della domanda: l'eccessivo utilizzo delle risorse può comportare una latenza elevata e una riduzione complessiva delle prestazioni. Il sottoutilizzo, invece, può comportare un sovradimensionamento delle risorse a fronte di un'ottimizzazione dei costi non ottimale. Dimensionando correttamente gli ambienti, è possibile trovare un equilibrio tra efficienza delle prestazioni e ottimizzazione dei costi. Per ovviare all'utilizzo eccessivo o insufficiente delle risorse, ElastiCache è dimensionabile in due dimensioni. È possibile dimensionare verticalmente aumentando o diminuendo la capacità del nodo. Puoi anche dimensionare orizzontalmente aggiungendo e rimuovendo nodi.

- [Obbligatorio] L'eccessivo utilizzo della CPU e della rete sui nodi primari deve essere risolto determinando e reindirizzando le operazioni di lettura ai nodi di replica. Usa i nodi di replica per le operazioni di lettura per ridurre l'utilizzo dei nodi primari. Puoi configurarli nella libreria client Redis connettendoti all'endpoint di lettura ElastiCache per la modalità cluster disabilitata o utilizzando il comando READONLY di Redis per la modalità cluster abilitata.

[Risorse]:

- [Individuazione degli endpoint di connessione](#)
- [Cluster Right-Sizing](#)
- [READONLY](#)
- [Obbligatorio] Monitora l'utilizzo delle risorse critiche del cluster come CPU, memoria e rete. L'utilizzo di queste risorse specifiche del cluster deve essere monitorato per determinare la decisione e il tipo di operazione di dimensionamento. Se la modalità cluster ElastiCache per Redis è disabilitata, i nodi primari e di replica possono essere dimensionati verticalmente. I nodi di replica possono anche dimensionare orizzontalmente da 0 a 5 nodi. Per la modalità cluster abilitata, lo stesso vale per ogni partizione del cluster. Inoltre, puoi aumentare o ridurre il numero di partizioni.

[Risorse]:

- [Monitoring best practices with Amazon ElastiCache for Redis using Amazon CloudWatch](#)
- [Dimensionamento ElastiCache per i cluster Redis](#)
- [Dimensionamento ElastiCache per i cluster Memcached](#)
- [Best practice] Il monitoraggio delle tendenze nel tempo può aiutarti a rilevare i cambiamenti del carico di lavoro che rimarrebbero inosservati se monitorati solo in un determinato momento. Per rilevare tendenze a lungo termine, utilizza le metriche di CloudWatch per analizzare intervalli di tempo più lunghi. I risultati derivanti dall'osservazione di periodi prolungati delle metriche di CloudWatch influenzano le previsioni sull'utilizzo delle risorse del cluster. I punti dati e le metriche di CloudWatch sono disponibili per un massimo di 455 giorni.

[Risorse]:

- [Monitoraggio dell'utilizzo con i parametri di CloudWatch](#)
- [Monitoring Memcached with CloudWatch Metrics](#)
- [Monitoring best practices with Amazon ElastiCache for Redis using Amazon CloudWatch](#)
- [Best practice] Se le risorse ElastiCache sono create con CloudFormation, è consigliabile eseguire modifiche utilizzando i modelli CloudFormation per preservare la coerenza operativa ed evitare ~~modifiche alla configurazione non gestite e spostamenti dello stack.~~

[Risorse]:

- [ElastiCache resource type reference for CloudFormation](#)
- [Best practice] Automatizza le operazioni di dimensionamento utilizzando i dati operativi del cluster e definisci soglie in CloudWatch per configurare gli allarmi. Usa Eventi CloudWatch e Simple Notification Service (SNS) per attivare le funzioni Lambda ed eseguire un'API ElastiCache per dimensionare automaticamente i cluster. Un esempio potrebbe essere aggiungere una partizione al cluster quando la metrica `EngineCPUUtilization` raggiunge l'80% per un lungo periodo di tempo. Un'altra opzione potrebbe essere utilizzare `DatabaseMemoryUsedPercentages` per una soglia basata sulla memoria.

[Risorse]:

- [Using Amazon CloudWatch Alarms](#)
- [Che cos'è Amazon CloudWatch Events?](#)
- [Utilizzo di AWS Lambda con Amazon SNS](#)
- [Informazioni di riferimento ElastiCache](#)

EO 3: come si gestiscono le risorse del cluster ElastiCache e si mantiene aggiornato il cluster?

Introduzione della domanda: quando si opera su larga scala, è essenziale essere in grado di individuare e identificare tutte le risorse ElastiCache. Quando si implementano nuove funzionalità dell'applicazione, è necessario creare una simmetria delle versioni del cluster in tutti i tipi di ambiente ElastiCache: sviluppo, test e produzione. Gli attributi delle risorse consentono di separare gli ambienti per diversi obiettivi operativi, ad esempio quando si implementano nuove funzionalità e si abilitano nuovi meccanismi di sicurezza.

Vantaggio della domanda: la separazione degli ambienti di sviluppo, test e produzione è una best practice operativa. È inoltre consigliabile che ai cluster e ai nodi in tutti gli ambienti vengano applicate le patch software più recenti utilizzando i processi appresi e documentati. Sfruttare le funzionalità native di ElastiCache consente al team di progettazione di concentrarsi sul raggiungimento degli obiettivi aziendali e non sulla manutenzione di ElastiCache.

- [Best practice] Esegui l'ultima versione del motore disponibile e applica gli aggiornamenti self-service non appena vengono rilasciati. ElastiCache aggiorna automaticamente l'infrastruttura sottostante durante la finestra di manutenzione specificata del cluster. Tuttavia, i nodi in esecuzione nei cluster vengono aggiornati tramite aggiornamenti self-service. Questi aggiornamenti

possono essere di due tipi: patch di sicurezza o aggiornamenti software secondari. Assicurati di comprendere la differenza tra i tipi di patch e quando vengono applicate.

[Risorse]:

- [Aggiornamenti di servizio in ElastiCache for Redis](#)
- [Pagina di assistenza agli aggiornamenti del servizio e alla manutenzione gestita da Amazon ElastiCache](#)
- [Best practice] Organizza le risorse ElastiCache utilizzando i tag. Usa i tag sui gruppi di replica e non sui singoli nodi. È possibile configurare i tag in modo che vengano visualizzati quando si eseguono query sulle risorse e utilizzare i tag per eseguire ricerche e applicare filtri. È consigliabile utilizzare i gruppi di risorse per creare e gestire facilmente le raccolte di risorse che condividono set di tag comuni.

[Risorse]:

- [Tagging Best Practices](#)
- [ElastiCache resource type reference for CloudFormation](#)
- [Gruppi di parametri](#)

EO 4: come si gestiscono le connessioni dei client ai cluster ElastiCache?

Introduzione della domanda: quando si opera su larga scala, è necessario capire in che modo i client si connettono al cluster ElastiCache per gestire gli aspetti operativi delle applicazioni (come i tempi di risposta).

Vantaggio della domanda: la scelta del meccanismo di connessione più appropriato garantisce che l'applicazione non si disconnetta a causa di errori di connettività, come i timeout.

- [Obbligatorio] Separa le operazioni di lettura da quelle di scrittura e connettiti ai nodi di replica per eseguire le operazioni di lettura. Tuttavia, tieni presente che quando separi le scritture dalle letture, perdi la capacità di leggere una chiave subito dopo averla scritta a causa della natura asincrona della replica Redis. Il comando WAIT può essere utilizzato per migliorare la sicurezza dei dati nel mondo reale e forzare le repliche a riconoscere le scritture prima di rispondere ai client, a un costo di prestazioni complessivo. L'utilizzo dei nodi di replica per le operazioni di lettura può essere configurato nella libreria client ElastiCache per Redis utilizzando l'endpoint di lettura ElastiCache per la modalità cluster disabilitata. Per la modalità cluster abilitata, utilizza il comando READONLY di ElastiCache per Redis. Per molte delle librerie client ElastiCache per Redis, il

comando READONLY di ElastiCache per Redis è implementato per impostazione predefinita o tramite un'impostazione di configurazione.

[Risorse]:

- [Individuazione degli endpoint di connessione](#)
- [READONLY](#)
- [Obbligatorio] Usa il pool di connessioni. La creazione di una connessione TCP comporta un costo in termini di tempo di CPU sia sul lato client che su quello server e il pool consente di riutilizzare la connessione TCP.

Per ridurre il sovraccarico della connessione, è necessario utilizzare il pool di connessioni. Con un pool di connessioni, l'applicazione può riutilizzare e rilasciare connessioni "secondo le necessità", senza il costo di stabilire la connessione. Puoi implementare il pool di connessioni tramite la libreria client ElastiCache per Redis (se supportata), con un framework disponibile per il tuo ambiente applicativo, oppure crearlo da zero.

- [Best practice] Assicurati che il timeout del socket del client sia impostato su almeno un secondo (rispetto al tipico valore predefinito "nessuno" in diversi client).
 - L'impostazione di un valore troppo basso può causare possibili timeout quando il carico del server è elevato. Se si imposta un valore troppo alto, l'applicazione può impiegare molto tempo per rilevare i problemi di connessione.
 - Controlla il volume delle nuove connessioni implementando il pool di connessioni nell'applicazione client. Ciò riduce la latenza e l'utilizzo della CPU necessari per aprire e chiudere le connessioni ed eseguire un handshake TLS se TLS è abilitato sul cluster.

[Risorse]: [Configure Amazon ElastiCache for Redis for higher availability](#)

- [Positivo] L'utilizzo delle pipeline (quando i casi d'uso lo consentono) può aumentare significativamente le prestazioni.
 - Con le pipeline si riduce il Round-Trip Time (RTT) tra i client delle applicazioni e il cluster e le nuove richieste possono essere elaborate anche se il client non ha ancora letto le risposte precedenti.
 - Con le pipeline puoi inviare più comandi al server senza attendere le risposte e le conferme. L'aspetto negativo delle pipeline è che quando alla fine recuperi tutte le risposte in blocco, potrebbe essere restituito un errore che non è riscontrabile fino alla fine.
 - Implementa i metodi per riprovare le richieste quando viene restituito un errore che omette la richiesta non valida.

[Risorse]: [Redis pipelining](#)

EO 5: come si distribuiscono i componenti ElastiCache per un carico di lavoro?

Introduzione della domanda: gli ambienti ElastiCache possono essere distribuiti manualmente tramite la console AWS o in modo programmatico tramite API, CLI, toolkit e così via. Le best practice dell'eccellenza operativa suggeriscono di automatizzare le implementazioni tramite il codice ogni volta che è possibile. Inoltre, i cluster ElastiCache possono essere isolati in base al carico di lavoro o combinati per scopi di ottimizzazione dei costi.

Vantaggio della domanda: la scelta del meccanismo di implementazione più appropriato per gli ambienti ElastiCache può migliorare l'eccellenza operativa nel tempo. Ti consigliamo di eseguire operazioni sotto forma di codice ogni volta che è possibile per ridurre al minimo l'errore umano e aumentare la ripetibilità, la flessibilità e i tempi di risposta agli eventi.

Comprendendo i requisiti di isolamento del carico di lavoro, puoi scegliere di avere ambienti ElastiCache dedicati per carico di lavoro o unire più carichi di lavoro in singoli cluster o in una combinazione di questi. Comprendere i compromessi può aiutare a trovare un equilibrio tra eccellenza operativa e ottimizzazione dei costi.

- [Obbligatorio] Comprendi le opzioni di implementazione disponibili per ElastiCache e automatizza queste procedure ogni volta che è possibile. I possibili modi di automazione includono CloudFormation, CLI/SDK AWS e API.

[Risorse]:

- [Amazon ElastiCache resource type reference](#)
- [elasticache](#)
- [Informazioni sull'API di Amazon ElastiCache](#)
- [Obbligatorio] Per tutti i carichi di lavoro, determina il livello di isolamento del cluster necessario.
 - [Best practice]: isolamento elevato, una mappatura 1:1 del carico di lavoro ai cluster. Consente un controllo granulare su accesso, dimensioni, dimensionamento e gestione delle risorse ElastiCache in base al carico di lavoro.
 - [Consigliato]: isolamento medio, M:1 isolato per scopo ma forse condiviso tra più carichi di lavoro (ad esempio un cluster dedicato alla memorizzazione nella cache dei carichi di lavoro e un altro dedicato alla messaggistica).

- [Positivo]: isolamento basso, M:1 tutti gli scopi e completamente condiviso. Consigliato per carichi di lavoro in cui è accettabile l'accesso condiviso.

EO 6: come si pianificano e si contengono gli errori?

Introduzione della domanda: l'eccellenza operativa include l'anticipazione degli errori eseguendo regolarmente esercitazioni "pre mortem" per identificare le potenziali fonti di errore in modo che possano essere rimosse o contenute. ElastiCache offre un'API di failover che consente di simulare eventi di errore dei nodi, a scopo di test.

Vantaggio della domanda: testando in anticipo gli scenari di errore, puoi scoprire in che modo influiscono sul tuo carico di lavoro. Ciò ti consente di testare in sicurezza le procedure di risposta e la loro efficacia, oltre a familiarizzare con l'esecuzione.

[Obbligatorio] Esegui regolarmente test di failover negli account di sviluppo/test. [TestFailover](#)

EO 7: come si risolvono gli eventi del motore Redis?

Introduzione della domanda: l'eccellenza operativa richiede la capacità di analizzare le informazioni a livello di servizio e a livello di motore per capire l'integrità e lo stato dei cluster. Amazon ElastiCache per Redis può inviare i log del motore Redis sia ad Amazon CloudWatch che ad Amazon Kinesis Data Firehose.

Vantaggio della domanda: l'attivazione dei log dei motori Redis sui cluster Amazon ElastiCache per Redis fornisce informazioni sugli eventi che influiscono sullo stato e sulle prestazioni dei cluster. I log del motore Redis forniscono dati direttamente dal motore Redis che non sono disponibili tramite il meccanismo degli eventi ElastiCache. Attraverso un'attenta osservazione degli eventi ElastiCache (vedi il passaggio precedente EO 1) e dei log del motore Redis, è possibile determinare un ordine di eventi per la risoluzione dei problemi sia dal punto di vista del servizio ElastiCache che del motore Redis.

- [Obbligatorio] Assicurati che sia abilitata la funzionalità di registrazione del motore Redis, disponibile in ElastiCache per Redis 6.2 e versioni successive. Questa operazione può essere eseguita durante la creazione del cluster o modificando il cluster dopo la creazione.
- Determina se File di log Amazon CloudWatch o Amazon Kinesis Data Firehose sono la destinazione appropriata per i log dei motori Redis.

- Seleziona un log di destinazione appropriato in CloudWatch o Kinesis Data Firehose per salvare in modo persistente i log. Se disponi di più cluster, considera un log di destinazione diverso per ogni cluster, in quanto ciò contribuisce a isolare i dati per la risoluzione dei problemi.

[Risorse]:

- Consegna dei registri: [Consegna dei registri](#)
- Destinazioni dei log: [File di log Amazon CloudWatch](#)
- Introduzione a File di log Amazon CloudWatch: [Che cos'è Amazon CloudWatch?](#)
- Introduzione ad Amazon Kinesis Data Firehose: [What Is Amazon Kinesis Data Firehose?](#)
- [Best practice] Se usi File di log Amazon CloudWatch, valuta la possibilità di utilizzare Logs Insights di Amazon CloudWatch per eseguire le query sul log del motore Redis per ottenere informazioni importanti.

Ad esempio, crea una query per il gruppo di log CloudWatch contenente i log del motore Redis che restituiscono eventi con un LogLevel pari a "WARNING", ad esempio:

```
fields @timestamp, LogLevel, Message
| sort @timestamp desc
| filter LogLevel = "WARNING"
```

[Risorse]: [Analisi dei dati di log con Amazon CloudWatch Logs Insights](#)

Pilastro della sicurezza di Amazon ElastiCache Well-Architected Lens

Il pilastro della sicurezza si concentra sulla protezione delle informazioni e dei sistemi. Gli argomenti chiave includono la riservatezza e l'integrità dei dati, l'identificazione e la gestione di chi può fare cosa mediante la gestione basata su privilegi, la protezione dei sistemi e l'istituzione di controlli per rilevare gli eventi di sicurezza.

Argomenti

- [SIC 1: quali sono le misure in uso per controllare l'accesso autorizzato ai dati di ElastiCache?](#)
- [SIC 2: le applicazioni richiedono per ElastiCache un'autorizzazione aggiuntiva oltre ai controlli basati sulla rete?](#)
- [SIC 3: esiste il rischio che i comandi possano essere eseguiti inavvertitamente, causando la perdita di dati o errori nei dati?](#)
- [SIC 4: come garantire la crittografia dei dati a riposo con ElastiCache](#)

- [SIC 5: come si crittografano i dati in transito con ElastiCache?](#)
- [SIC 6: come si limita l'accesso alle risorse del piano di controllo \(control-plane\)?](#)
- [SIC 7: come si rileva e si risponde agli eventi di sicurezza?](#)

SIC 1: quali sono le misure in uso per controllare l'accesso autorizzato ai dati di ElastiCache?

Introduzione della domanda: tutti i cluster ElastiCache sono progettati per essere accessibili da istanze Amazon Elastic Compute Cloud in un VPC, funzioni serverless (AWS Lambda) o container (Amazon Elastic Container Service). Lo scenario più comune è accedere a un cluster ElastiCache da un'istanza Amazon Elastic Compute Cloud nello stesso Amazon Virtual Private Cloud (Amazon VPC). Prima di poter eseguire la connessione a un cluster da un'istanza Amazon EC2, devi autorizzare l'istanza Amazon EC2 ad accedere al cluster. Per accedere a un cluster ElastiCache in esecuzione in un VPC, è necessario fornire l'ingresso della rete nel cluster.

Vantaggio della domanda: l'ingresso della rete nel cluster è controllato tramite i gruppi di sicurezza del VPC. Un gruppo di sicurezza funge da firewall virtuale per le istanze Amazon EC2 per controllare il traffico in entrata e quello in uscita. Le regole in entrata controllano il traffico in entrata verso l'istanza e le regole in uscita controllano il traffico in uscita dall'istanza. Nel caso di ElastiCache, quando si avvia un cluster, è necessario associare un gruppo di sicurezza. In tal modo si garantisce che le regole del traffico in entrata e in uscita siano in atto per tutti i nodi che costituiscono il cluster. Inoltre, ElastiCache è configurato per implementare esclusivamente su sottoreti private, in modo da essere accessibile solo tramite la rete privata del VPC.

- [Obbligatorio] Il gruppo di sicurezza associato al cluster controlla l'ingresso e l'accesso della rete al cluster. Per impostazione predefinita, un gruppo di sicurezza non ha alcuna regola in entrata definita e, pertanto, nessun percorso di ingresso a ElastiCache. Per abilitarlo, configura una regola in entrata per il gruppo di sicurezza che specifichi l'indirizzo/intervallo IP di origine, il traffico di tipo TCP e la porta del cluster ElastiCache (ad esempio, la porta predefinita 6379 per ElastiCache per Redis). Sebbene sia possibile consentire un set molto ampio di origini di ingresso, come tutte le risorse all'interno di un VPC (0.0.0.0/0), si consiglia di essere il più granulari possibile nella definizione delle regole in entrata, ad esempio autorizzare solo l'accesso in entrata ai client Redis in esecuzione su istanze Amazon EC2 associate a uno specifico gruppo di sicurezza.

[Risorse]:

- [Sottoreti e gruppi di sottoreti](#)
- [Accesso al cluster o gruppo di replica](#)

- [Controlla il traffico verso le risorse utilizzando gruppi di sicurezza](#)
- [Gruppi di sicurezza Amazon EC2 per istanze Linux](#)
- [Obbligatorio] Le policy AWS Identity and Access Management possono essere assegnate a funzioni AWS Lambda che possono accedere ai dati ElastiCache. Per abilitare questa funzionalità, crea un ruolo di esecuzione IAM con l'autorizzazione `AWSLambdaVPCAccessExecutionRole`, quindi assegna il ruolo alla funzione AWS Lambda.

[Risorse]: Configurazione di una funzione Lambda per l'accesso a Amazon ElastiCache in un Amazon VPC: [Tutorial: Configurazione di una funzione Lambda per l'accesso a Amazon ElastiCache in un Amazon VPC](#)

SIC 2: le applicazioni richiedono per ElastiCache un'autorizzazione aggiuntiva oltre ai controlli basati sulla rete?

Introduzione della domanda: negli scenari in cui è necessario limitare o controllare l'accesso ai cluster ElastiCache per Redis a livello di singolo client, si consiglia di eseguire l'autenticazione tramite il comando AUTH di ElastiCache per Redis. I token di autenticazione ElastiCache per Redis, con gestione opzionale di utenti e gruppi di utenti, consentono a ElastiCache per Redis di richiedere una password prima di consentire ai client di eseguire comandi e chiavi di accesso, migliorando così la sicurezza del piano dati.

Vantaggio della domanda: per una migliore sicurezza dei dati, ElastiCache per Redis fornisce meccanismi di protezione dall'accesso non autorizzato ai dati, tra cui l'esecuzione del comando AUTH del controllo degli accessi basato su ruoli (RBAC) o del token AUTH (password) da parte dei client per connettersi a ElastiCache prima di eseguire comandi autorizzati.

- [Best practice] Per ElastiCache per Redis 6.x e versioni successive, determina i controlli di autenticazione e autorizzazione definendo gruppi di utenti, utenti e stringhe di accesso. Assegna gli utenti ai gruppi di utenti, quindi assegna i gruppi di utenti ai cluster. Per utilizzare il RBAC, è necessario selezionarlo al momento della creazione del cluster e abilitare la crittografia in transito. Assicurati di usare un client Redis che supporti TLS per poter utilizzare RBAC.

[Risorse]:

- [Applicazione di RBAC a un gruppo di replica per ElastiCache for Redis](#)
- [Specifica delle autorizzazioni mediante una stringa di accesso](#)
- [ACL](#)

- [ElastiCache supportate per le versioni Redis](#)
- [Best practice] Per le versioni di ElastiCache per Redis precedenti alla 6.x, oltre a impostare una password/token complessa e mantenere una policy rigorosa per le password per il comando AUTH di ElastiCache per Redis, è consigliabile ruotare la password/token. ElastiCache può gestire fino a due (2) token di autenticazione in qualsiasi momento. Puoi anche modificare il cluster per richiedere esplicitamente l'uso di token di autenticazione.

[Risorse]: [Modifica del token AUTH su un cluster ElastiCache per Redis esistente](#)

SIC 3: esiste il rischio che i comandi possano essere eseguiti inavvertitamente, causando la perdita di dati o errori nei dati?

Introduzione della domanda: esistono numerosi comandi Redis che possono avere impatti negativi sulle operazioni se eseguiti per errore o da malintenzionati. Questi comandi possono avere conseguenze impreviste dal punto di vista delle prestazioni e della sicurezza dei dati. Ad esempio, uno sviluppatore che esegue regolarmente il comando FLUSHALL in un ambiente di sviluppo potrebbe per errore chiamare inavvertitamente questo comando in un sistema di produzione, con conseguente perdita accidentale di dati.

Vantaggio della domanda: a partire da ElastiCache per Redis 5.0.3, è possibile su ElastiCache rinominare determinati comandi che potrebbero disturbare il carico di lavoro. La ridenominazione dei comandi può aiutare a evitare che vengano eseguiti inavvertitamente sul cluster.

- [Obbligatorio]

[Risorse]:

- [ElastiCache per Redis versione 5.0.3 \(obsoleta, usa versione 5.0.6\)](#)
- [Modifiche dei parametri di Redis 5.0.3](#)
- [Redis security](#)

SIC 4: come garantire la crittografia dei dati a riposo con ElastiCache

Introduzione della domanda: sebbene ElastiCache per Redis sia un archivio di dati in memoria, è possibile crittografare tutti i dati che possono essere conservati (nello spazio di archiviazione) come parte delle operazioni standard del cluster. Ad esempio i backup pianificati e manuali scritti su Amazon S3, nonché i dati salvati nello spazio di archiviazione su disco a seguito di operazioni di

sincronizzazione e scambio. I tipi di istanza delle famiglie M6g e R6g offrono anche la crittografia in memoria sempre attiva.

Vantaggio della domanda: ElastiCache per Redis offre una crittografia opzionale dei dati a riposo per aumentare la sicurezza.

- [Obbligatorio] La crittografia dei dati a riposo può essere abilitata su un cluster ElastiCache (gruppo di replica) solo al momento della creazione. Un cluster esistente non può essere modificato per iniziare a crittografare i dati a riposo. Per impostazione predefinita, ElastiCache fornisce e gestisce le chiavi utilizzate nella crittografia a riposo.

[Risorse]:

- [Condizioni di crittografia dei dati inattivi](#)
- [Abilitazione della crittografia dei dati inattivi](#)
- [Best practice] Utilizza i tipi di istanza Amazon EC2 che crittografano i dati mentre sono in memoria (come M6g o R6g). Ove possibile, valuta la possibilità di gestire le chiavi per la crittografia a riposo. Per gli ambienti di sicurezza dei dati più rigorosi è possibile utilizzare AWS Key Management Service (KMS) per gestire autonomamente le chiavi master del cliente (CMK). Grazie all'integrazione di ElastiCache con AWS Key Management Service, puoi creare, impostare e gestire le chiavi utilizzate per la crittografia dei dati a riposo per il cluster ElastiCache per Redis.

[Risorse]:

- [Utilizzo di chiavi gestite dal cliente da AWS Key Management Service](#)
- [AWS Key Management Service](#)
- [AWS KMS concepts](#)

SIC 5: come si crittografano i dati in transito con ElastiCache?

Introduzione della domanda: è un requisito comune per evitare che i dati vengano compromessi durante il transito. Rappresenta i dati all'interno dei componenti di un sistema distribuito, nonché tra client applicativi e nodi del cluster. ElastiCache per Redis supporta questo requisito consentendo la crittografia dei dati in transito tra client e cluster e tra i nodi del cluster stessi. I tipi di istanza delle famiglie M6g e R6g offrono anche la crittografia in memoria sempre attiva.

Vantaggio della domanda: la crittografia in transito di Amazon ElastiCache è una funzionalità opzionale che ti consente di aumentare la sicurezza dei tuoi dati nei punti più vulnerabili, quando sono in transito da una posizione all'altra.

- [Obbligatorio] La crittografia in transito può essere abilitata solo su un cluster ElastiCache per Redis (gruppo di replica) al momento della creazione. Tieni presente che, a causa dell'elaborazione aggiuntiva richiesta per la crittografia/decrittografia dei dati, l'implementazione della crittografia in transito avrà un certo impatto sulle prestazioni. Per comprendere l'impatto, ti consigliamo di confrontare il carico di lavoro prima e dopo aver abilitato la crittografia in transito.

[Risorse]:

- [Panoramica della crittografia dei dati in transito](#)

SIC 6: come si limita l'accesso alle risorse del piano di controllo (control-plane)?

Introduzione della domanda: le policy IAM e l'ARN consentono controlli di accesso granulari per ElastiCache per Redis, abilitando un controllo più rigoroso per gestire la creazione, la modifica e l'eliminazione dei cluster ElastiCache per Redis.

Vantaggio della domanda: la gestione delle risorse di Amazon ElastiCache, come gruppi di replica, nodi e così via può essere limitata agli account AWS con autorizzazioni specifiche basate sulle policy IAM, migliorando la sicurezza e l'affidabilità delle risorse.

- [Obbligatorio] Gestisci l'accesso alle risorse di Amazon ElastiCache assegnando agli utenti AWS policy AWS Identity and Access Management specifiche che consentono un controllo più preciso sulle azioni che gli account possono eseguire sui cluster.

[Risorse]:

- [Panoramica della gestione delle autorizzazioni di accesso alle risorse di ElastiCache](#)
- [Utilizzo di policy basate su identità \(policy IAM\) per Amazon ElastiCache](#)

SIC 7: come si rileva e si risponde agli eventi di sicurezza?

Introduzione della domanda: ElastiCache, se implementato con il RBAC abilitato, esporta i parametri di CloudWatch per notificare agli utenti gli eventi di sicurezza. Queste metriche aiutano a identificare i tentativi di autenticazione non riusciti, le chiavi di accesso o l'esecuzione di comandi per i quali la connessione degli utenti con il RBAC non è autorizzata.

Inoltre, le risorse relative a prodotti e servizi AWS aiutano a proteggere il carico di lavoro complessivo automatizzando le implementazioni e registrando tutte le azioni e le modifiche per una revisione o un controllo successivo.

Vantaggio della domanda: monitorando gli eventi, consenti all'organizzazione di rispondere in base a requisiti, policy e procedure. L'automazione del monitoraggio e delle risposte a questi eventi rafforza il livello generale di sicurezza.

- [Obbligatorio] Acquisisci familiarità con i parametri di CloudWatch pubblicati relativi agli errori di autenticazione e autorizzazione del RBAC.
 - `AuthenticationFailures` = Tentativi di autenticazione non riusciti su Redis
 - `KeyAuthorizationFailures` = Tentativi degli utenti di accedere alle chiavi senza autorizzazione non riusciti
 - `CommandAuthorizationFailures` = Tentativi degli utenti di eseguire comandi senza autorizzazione non riusciti

[Risorse]:

- [Parametri di Redis](#)
- [Best practice] Ti consigliamo di configurare avvisi e notifiche su queste metriche e rispondere se necessario.

[Risorse]:

- [Utilizzo degli allarmi di Amazon CloudWatch](#)
- [Best practice] Usa il comando `ACL LOG` di Redis per raccogliere ulteriori dettagli

[Risorse]:

- [ACL LOG](#)
- [Best practice] Acquisisci familiarità con le funzionalità dei prodotti e dei servizi AWS per quanto riguarda il monitoraggio, la registrazione e l'analisi delle implementazioni e degli eventi di ElastiCache

[Risorse]:

- [Registrazione delle chiamate all'API Amazon ElastiCache con AWS CloudTrail](#)
- [elasticache-redis-cluster-automatic-backup-check](#)
- [Monitoraggio dell'utilizzo con i parametri di CloudWatch](#)

Pilastro dell'affidabilità di Amazon ElastiCache Well-Architected Lens

Argomenti

- [AFF 1: in che modo si supportano le implementazioni di architetture ad alta disponibilità?](#)
- [AFF 2: in che modo si raggiungono gli obiettivi del punto di ripristino \(RPO\) con ElastiCache?](#)
- [AFF 3: come si supportano i requisiti di ripristino di emergenza?](#)
- [AFF 4: come si pianificano i failover in modo efficace?](#)
- [AFF 5: i tuoi componenti ElastiCache sono progettati per il dimensionamento?](#)

AFF 1: in che modo si supportano le implementazioni di architetture ad alta disponibilità?

Introduzione della domanda: l'architettura ad alta disponibilità di Amazon ElastiCache ti consente di operare in uno stato resiliente durante gli eventi di disponibilità.

Vantaggio della domanda: l'architettura dei cluster ElastiCache resilienti ai guasti garantisce una maggiore disponibilità per le implementazioni ElastiCache.

- [Obbligatorio] Determina il livello di affidabilità richiesto per il cluster ElastiCache in uso. Carichi di lavoro diversi hanno standard di resilienza diversi, da quelli totalmente effimeri a quelli mission critical. Definisci le esigenze per ogni tipo di ambiente in cui gestisci, ad esempio, sviluppo, test e produzione.

Motore di cache: Memcached ed ElastiCache per Redis

1. Memcached non fornisce alcun meccanismo di replica e viene utilizzato principalmente per carichi di lavoro effimeri.
 2. ElastiCache per Redis offre le funzionalità per l'alta disponibilità descritte di seguito
- [Best practice] Per i carichi di lavoro che richiedono alta disponibilità, usa ElastiCache per Redis in modalità cluster con un minimo di due repliche per partizione, anche per carichi di lavoro con requisiti di velocità di trasmissione effettiva contenuti che richiedono una sola partizione.
 1. Con la modalità cluster abilitata, multi-AZ viene impostato automaticamente.

Multi-AZ riduce al minimo i tempi di inattività eseguendo failover automatici dal nodo primario alle repliche, in caso di manutenzione pianificata o non pianificata, mitigando i guasti delle zone di disponibilità.

2. Per i carichi di lavoro a partizioni, un minimo di tre partizioni garantisce un ripristino più rapido durante gli eventi di failover, poiché il protocollo per i cluster Redis richiede la disponibilità della maggior parte dei nodi primari per raggiungere il quorum.
3. Configura due o più repliche per la disponibilità.

La presenza di due repliche offre una migliore scalabilità di lettura e anche la disponibilità di lettura in scenari in cui una replica è in fase di manutenzione.

4. Usa i tipi di nodi basati su Graviton2 (nodi predefiniti nella maggior parte delle regioni).

Amazon ElastiCache per Redis offre prestazioni ottimizzate per questi nodi. Di conseguenza, si ottengono migliori prestazioni di replica e sincronizzazione, con conseguente maggiore disponibilità complessiva.

5. Monitora e definisci le dimensioni corrette per gestire i picchi di traffico previsti. In caso di carichi elevati, il motore ElastiCache per Redis potrebbe non rispondere, influenzando sulla disponibilità. `BytesUsedForCache` e `DatabaseMemoryUsagePercentage` sono buoni indicatori dell'utilizzo della memoria, mentre `ReplicationLag` è un indicatore dello stato della replica in base alla velocità di scrittura. Puoi utilizzare queste metriche per attivare il dimensionamento dei cluster.
6. Garantisci la resilienza lato client eseguendo test con l'[API di failover prima di un evento di failover di produzione](#).

[Risorse]:

- [Configure Amazon ElastiCache for Redis for higher availability](#)
- [Alta disponibilità utilizzando gruppi di replica](#)

AFF 2: in che modo si raggiungono gli obiettivi del punto di ripristino (RPO) con ElastiCache?

Introduzione della domanda: comprendi l'obiettivo del punto di ripristino del carico di lavoro per prendere decisioni informate sulle strategie di backup e ripristino di ElastiCache.

Vantaggio della domanda: disporre di una strategia RPO in atto può migliorare la continuità aziendale in caso di scenari di ripristino di emergenza. La progettazione delle policy di backup e ripristino può aiutarti a raggiungere i tuoi obiettivi del punto di ripristino (RPO) per i dati ElastiCache. ElastiCache per Redis offre funzionalità di snapshot archiviate in Amazon S3, insieme a una policy di conservazione configurabile. Questi snapshot vengono acquisiti durante la finestra di backup definita e gestiti automaticamente dal servizio. Se il carico di lavoro richiede una maggiore granularità del backup, hai la possibilità di creare fino a 20 backup manuali al giorno. I backup creati manualmente non hanno una policy di conservazione del servizio e possono essere conservati a tempo indeterminato.

- [Obbligatorio] Comprendi e documenta l'obiettivo del punto di ripristino delle implementazioni ElastiCache.
 - Tieni presente che Memcached non offre processi di backup.
 - Esamina le funzionalità di backup e ripristino di ElastiCache.
- [Best practice] Predisponi di una procedura di comunicazione per il backup del cluster.
 - Avvia i backup manuali in base alle necessità.
 - Esamina le policy di conservazione per i backup automatici.
 - Tieni presente che i backup manuali vengono conservati a tempo indeterminato.
 - Pianifica i backup automatici nei periodi di basso utilizzo.
 - Esegui operazioni di backup su repliche di lettura per ridurre al minimo l'impatto sulle prestazioni del cluster.
- [Positivo] Sfrutta la funzionalità di backup pianificato di ElastiCache per eseguire regolarmente il backup dei dati durante una finestra definita.
 - Esegui periodicamente il test del ripristino dei tuoi backup.
- [Risorse]:
 - [Redis](#)
 - [Backup e ripristino per ElastiCache for Redis](#)
 - [Creazione di backup manuali](#)
 - [Pianificazione di backup automatici](#)
 - [Backup and Restore ElastiCache Redis Clusters](#)

AFF 3: come si supportano i requisiti di ripristino di emergenza?

Introduzione della domanda: il ripristino di emergenza è un aspetto importante di qualsiasi pianificazione del carico di lavoro. ElastiCache per Redis offre diverse opzioni per implementare il ripristino di emergenza in base ai requisiti di resilienza del carico di lavoro. Con Global Datastore di Amazon ElastiCache per Redis puoi scrivere sul tuo cluster ElastiCache per Redis in una regione e avere i dati disponibili per la lettura da altri due cluster di replica interregionali, consentendo così letture a bassa latenza e ripristino di emergenza tra le regioni.

Vantaggio della domanda: la comprensione e la pianificazione di una varietà di scenari di emergenza possono garantire la continuità aziendale. Le strategie di ripristino di emergenza devono essere ~~bilanciate rispetto ai costi, all'impatto sulle prestazioni e alla potenziale perdita di dati.~~

- [Obbligatorio] Sviluppa e documenta strategie di ripristino di emergenza per tutti i componenti ElastiCache in base ai requisiti del carico di lavoro. ElastiCache è unico in quanto alcuni casi d'uso sono del tutto effimeri e non richiedono alcuna strategia di ripristino di emergenza, mentre altri sono all'estremità opposta dello spettro e richiedono una strategia di ripristino di emergenza estremamente robusta. Tutte le opzioni devono essere valutate rispetto all'ottimizzazione dei costi: una maggiore resilienza richiede una maggiore quantità di infrastruttura.

Comprendi le opzioni di ripristino di emergenza disponibili a livello regionale e multiregionale.

- Le implementazioni multi-AZ sono consigliate per evitare errori di zone di disponibilità. Assicurati di eseguire l'implementazione con la modalità cluster abilitata nelle architetture multi-AZ, con un minimo di 3 zone di disponibilità.
- Global Datastore è consigliato per proteggersi dagli errori a livello di regione.
- [Best practice] Abilita Global Datastore per i carichi di lavoro che richiedono resilienza a livello di regione.
 - Prepara un piano di failover nella regione secondaria in caso di degrado di quella primaria.
 - Esegui il test del processo di failover multiregione prima di eseguire un failover in produzione.
 - Monitora la metrica `ReplicationLag` per comprendere il potenziale impatto della perdita di dati durante gli eventi di failover.
- [Risorse]:
 - [Limitazione dell'impatto degli errori](#)
 - [Replica tra regioni AWS mediante Global Datastore](#)
 - [Ripristino da un backup con ridimensionamento cluster opzionale](#)
 - [Riduzione dei tempi di inattività in ElastiCache for Redis con Multi-AZ](#)

AFF 4: come si pianificano i failover in modo efficace?

Introduzione della domanda: l'abilitazione di multi-AZ con failover automatici è una best practice di ElastiCache. In alcuni casi, ElastiCache per Redis sostituisce i nodi primari come parte delle operazioni di servizio. Ad esempio nel caso di eventi di manutenzione programmata e nel caso poco probabile di un errore in un nodo o una zona di disponibilità. I failover ottimali si basano su ElastiCache e sulla configurazione della libreria client.

Vantaggio della domanda: seguire le best practice per i failover ElastiCache in combinazione con la libreria client specifica di ElastiCache per Redis consente di ridurre al minimo i potenziali tempi di inattività durante gli eventi di failover.

- [Obbligatorio] Per la modalità cluster disabilitata, utilizza i timeout in modo che i client rilevino se è necessario disconnettersi dal vecchio nodo primario e riconnettersi al nuovo nodo primario, utilizzando l'indirizzo IP dell'endpoint primario aggiornato. Per la modalità cluster abilitata, la libreria client è responsabile del rilevamento delle modifiche nella topologia del cluster sottostante. Questa operazione viene spesso eseguita tramite le impostazioni di configurazione nella libreria client ElastiCache per Redis, che consentono anche di configurare la frequenza e il metodo di aggiornamento. Ogni libreria client offre le proprie impostazioni e maggiori dettagli sono disponibili nella documentazione corrispondente.

[Risorse]:

- [Riduzione dei tempi di inattività in ElastiCache for Redis con Multi-AZ](#)
- Esamina le best practice della tua libreria client ElastiCache per Redis.
- [Obbligatorio] Il successo dei failover dipende da un ambiente di replica integro tra il nodo primario e quello di replica. Esamina e comprendi la natura asincrona della replica Redis e le metriche CloudWatch disponibili per segnalare il ritardo di replica tra il nodo primario e quello di replica. Per i casi d'uso che richiedono una maggiore sicurezza dei dati, utilizza il comando WAIT di Redis per forzare le repliche a riconoscere le scritture prima di rispondere ai client connessi.

[Risorse]:

- [Parametri di Redis](#)
- [Monitoring best practices with Amazon ElastiCache for Redis using Amazon CloudWatch](#)
- [Best practice] Convalida regolarmente la reattività dell'applicazione durante il failover utilizzando l'API Test Failover di ElastiCache.

[Risorse]:

- [Testing Automatic Failover to a Read Replica on Amazon ElastiCache for Redis](#)
- [Test del failover automatico](#)

AFF 5: i tuoi componenti ElastiCache sono progettati per il dimensionamento?

Introduzione della domanda: comprendendo le funzionalità di dimensionamento e le topologie di implementazione disponibili, i componenti ElastiCache possono adattarsi nel tempo per soddisfare i requisiti del carico di lavoro in continuo cambiamento. ElastiCache offre un dimensionamento in 4 direzioni: in/out (orizzontale) e up/down (verticale).

Vantaggio della domanda: seguire le best practice per le implementazioni di ElastiCache offre la massima flessibilità di dimensionamento, oltre a soddisfare il principio Well-Architected del dimensionamento orizzontale per ridurre al minimo l'impatto degli errori.

- [Obbligatorio] Comprendi la differenza tra le topologie modalità cluster abilitata e modalità cluster disabilitata. In quasi tutti i casi è consigliabile eseguire l'implementazione con la modalità cluster abilitata in quanto consente una maggiore scalabilità nel tempo. I componenti in modalità cluster disabilitata sono limitati nella capacità di dimensionarsi orizzontalmente per aggiungere repliche di lettura.
- [Obbligatorio] Determina quando e come dimensionare.
 - Per ulteriori READIOPS: aggiungi repliche
 - Per ulteriori WRITEOPS: aggiungi partizioni (dimensionamento orizzontale)
 - Per ulteriori I/O di rete: utilizza istanze ottimizzate per la rete (dimensionamento verticale)
- [Best practice] Implementa i componenti ElastiCache con la modalità cluster abilitata, preferendo tanti piccoli nodi a pochi nodi grandi. In tal modo limiti efficacemente il raggio di applicazione dell'errore di un nodo.
- [Best practice] Includi le repliche nei cluster per una maggiore reattività durante gli eventi di dimensionamento
- [Positivo] Se la modalità cluster è disabilitata, utilizza le repliche di lettura per aumentare la capacità di lettura complessiva. ElastiCache supporta fino a 5 repliche di lettura in modalità cluster disabilitata, oltre al dimensionamento verticale.
- [Risorse]:
 - [Dimensionamento ElastiCache per i cluster Redis](#)
 - [Dimensionamento verso l'alto online](#)
 - [Dimensionamento ElastiCache per i cluster Memcached](#)

Pilastro dell'efficienza delle prestazioni di Amazon ElastiCache Well-Architected Lens

Il pilastro dell'efficienza delle prestazioni si concentra sull'uso efficiente delle risorse IT e di calcolo. Gli argomenti chiave includono la selezione delle dimensioni e dei tipi corretti per le risorse in base ai requisiti del carico di lavoro, il monitoraggio delle prestazioni e il processo per prendere decisioni informate e mantenere l'efficienza man mano che le esigenze aziendali cambiano.

Argomenti

- [EP 1: come si monitorano le prestazioni del cluster Amazon ElastiCache?](#)
- [EP 2: come si distribuisce il lavoro tra i nodi del cluster ElastiCache?](#)
- [EP 3: come si monitora e si segnala l'efficacia e le prestazioni della cache per i carichi di lavoro con memorizzazione nella cache?](#)
- [EP 4: in che modo il carico di lavoro ottimizza l'uso delle risorse e delle connessioni di rete?](#)
- [EP 5: come si gestisce l'eliminazione e/o l'espulsione delle chiavi?](#)
- [EP 6: come si modellano e si usano i dati in ElastiCache?](#)
- [EP 7: come si registrano i comandi a esecuzione lenta nel cluster Amazon ElastiCache?](#)
- [EP 8: in che modo il dimensionamento automatico aiuta ad aumentare le prestazioni del cluster ElastiCache?](#)

EP 1: come si monitorano le prestazioni del cluster Amazon ElastiCache?

Introduzione della domanda: comprendendo le metriche di monitoraggio esistenti è possibile determinare l'utilizzo corrente. Un monitoraggio adeguato può aiutare a individuare i potenziali ostacoli che influiscono sulle prestazioni del cluster.

Vantaggio della domanda: la comprensione delle metriche associate al cluster può aiutare nella definizione delle tecniche di ottimizzazione volte a conseguire la riduzione della latenza e l'aumento della velocità di trasmissione effettiva.

- [Obbligatorio] Test delle prestazioni di base utilizzando un sottoinsieme del carico di lavoro.
 - È necessario monitorare le prestazioni del carico di lavoro effettivo utilizzando meccanismi come i test di carico.
 - Monitora le metriche di CloudWatch durante l'esecuzione di questi test per comprendere i valori restituiti e stabilire la linea di base delle prestazioni.
- [Best practice] Per i carichi di lavoro di ElastiCache per Redis, rinomina i comandi costosi in termini di calcolo, ad esempio KEYS, per ridurre la possibilità che gli utenti eseguano comandi che bloccano i cluster di produzione.
 - I carichi di lavoro di ElastiCache per Redis che eseguono il motore 6.x possono utilizzare il controllo degli accessi basato su ruoli per limitare determinati comandi. L'accesso ai comandi può essere controllato creando utenti e gruppi di utenti con la console AWS o la CLI e quindi associandoli a un cluster ElastiCache per Redis. Quando il controllo degli accessi basato su ruoli

è abilitato in Redis 6, è possibile usare "-@dangerous" per impedire all'utente l'esecuzione di comandi costosi come KEYS, MONITOR, SORT e così via.

- Per la versione 5.x del motore, rinomina i comandi utilizzando il parametro `rename-commands` nel gruppo di parametri del cluster Amazon ElastiCache per Redis.
- [Consigliato] Analizza le query lente ed esamina le tecniche di ottimizzazione.
 - Per i carichi di lavoro di ElastiCache per Redis puoi analizzare meglio le tue query usando il log delle query lente. Ad esempio, puoi utilizzare il seguente comando `redis-cli slowlog get 10` per mostrare gli ultimi 10 comandi che hanno superato la soglia di latenza (10 secondi per impostazione predefinita).
 - Alcune query possono essere eseguite in modo più efficiente utilizzando le strutture di dati complesse di ElastiCache per Redis. Ad esempio, per le ricerche con intervalli di numeri, è possibile implementare nell'applicazione semplici indici numerici con i set ordinati. La gestione di questi indici può ridurre le scansioni eseguite sui set e restituire i dati con prestazioni migliori.
 - Per i carichi di lavoro di ElastiCache per Redis, `redis-benchmark` fornisce una semplice interfaccia per testare le prestazioni di diversi comandi utilizzando gli input definiti dall'utente, come il numero di client e la dimensione dei dati.
 - Poiché Memcached supporta solo semplici comandi a livello di chiave, valuta la possibilità di creare altre chiavi come indici per evitare l'iterazione dello spazio delle chiavi per rispondere alle query dei client.
- [Risorse]:
 - [Monitoraggio dell'utilizzo con i parametri di CloudWatch](#)
 - [Monitoraggio dell'utilizzo con i parametri di CloudWatch](#)
 - [Utilizzo degli allarmi di Amazon CloudWatch](#)
 - [Parametri specifici di Redis](#)
 - [SLOWLOG](#)
 - [Redis benchmark](#)

EP 2: come si distribuisce il lavoro tra i nodi del cluster ElastiCache?

Introduzione della domanda: il modo in cui l'applicazione si connette ai nodi Amazon ElastiCache può influire sulle prestazioni e sulla scalabilità del cluster.

Vantaggio della domanda: l'uso corretto dei nodi disponibili nel cluster garantisce la distribuzione del lavoro tra le risorse disponibili. Le seguenti tecniche consentono anche di evitare risorse inutilizzate.

- [Obbligatorio] I client devono connettersi all'endpoint ElastiCache appropriato.
 - Amazon ElastiCache per Redis implementa diversi endpoint in base alla modalità cluster in uso. Se la modalità cluster è abilitata, ElastiCache fornisce un endpoint di configurazione. Se la modalità cluster è disabilitata, ElastiCache fornisce un endpoint primario, in genere utilizzato per le scritture, e un endpoint di lettura per bilanciare le letture tra le repliche. L'implementazione corretta di questi endpoint si traduce in prestazioni migliori e operazioni di dimensionamento più semplici. Evita di connetterti ai singoli endpoint dei nodi a meno che non vi sia un requisito specifico in tal senso.
 - Per i cluster Memcached con più nodi, ElastiCache fornisce un endpoint di configurazione che abilita l'individuazione automatica. Ti consigliamo di utilizzare un algoritmo di hash per distribuire il lavoro in modo uniforme tra i nodi di cache. Molte librerie client Memcached implementano l'hash in modo coerente. Consulta la documentazione della libreria che utilizzi per verificare se supporta l'hashing coerente e ottenere informazioni su come implementarlo. Ulteriori informazioni sull'implementazione di queste funzionalità sono disponibili [qui](#).
- [Consigliato] Implementa una strategia per identificare e correggere le chiavi hot nel tuo carico di lavoro.
 - Considera l'impatto delle strutture di dati Redis multidimensionali come elenchi, flussi, set e così via. Queste strutture di dati sono archiviate in singole chiavi Redis che risiedono in un solo nodo. Una chiave multidimensionale molto grande può utilizzare più memoria e capacità di rete rispetto ad altri tipi di dati e causare un uso sproporzionato del nodo. Se possibile, progetta il tuo carico di lavoro in modo da distribuire l'accesso ai dati tra molte chiavi discrete.
 - Le chiavi hot del carico di lavoro possono influire sulle prestazioni del nodo in uso. Nei carichi di lavoro di ElastiCache per Redis, è possibile rilevare le chiavi hot utilizzando `redis-cli --hotkeys` se è in uso una policy LFU per la memoria massima.
 - Prendi in considerazione la replica delle chiavi hot su più nodi per distribuire l'accesso in modo più uniforme. Con questo approccio il client scrive su più nodi primari (il nodo Redis non fornisce questa funzionalità) e mantiene un elenco di nomi di chiavi da leggere, oltre al nome della chiave originale.
 - ElastiCache per Redis versione 6 supporta la [memorizzazione nella cache lato client](#) assistita dal server. In tal modo le applicazioni possono attendere le modifiche apportate alle chiavi prima di effettuare le chiamate di rete a ElastiCache.
- [Risorse]:
 - [Configure Amazon ElastiCache for Redis for higher availability](#)
 - [Individuazione degli endpoint di connessione](#)

- [Best practice per il bilanciamento del carico](#)
- [Client-side caching in Redis](#)

EP 3: come si monitora e si segnala l'efficacia e le prestazioni della cache per i carichi di lavoro con memorizzazione nella cache?

Introduzione della domanda: la memorizzazione nella cache è un carico di lavoro comune in ElastiCache ed è importante capire come gestire l'efficacia e le prestazioni della cache.

Vantaggio della domanda: l'applicazione potrebbe mostrare segni di rallentamento delle prestazioni. La capacità di utilizzare metriche specifiche della cache per decidere come aumentare le prestazioni delle app è fondamentale per il carico di lavoro della cache.

- [Obbligatorio] Misura e monitora nel tempo il rapporto dei riscontri della cache. L'efficienza della cache è determinata dal "rapporto di riscontri della cache". Il rapporto di riscontri della cache è definito dal totale dei riscontri delle chiavi diviso per il totale dei riscontri e dei mancati riscontri. Più il rapporto è vicino a 1, più efficace è la cache. Un basso rapporto di riscontri della cache è causato dal volume di mancati riscontri della cache. I mancati riscontri della cache si verificano quando la chiave richiesta non viene trovata nella cache. Una chiave non è nella cache perché è stata rimossa o eliminata, è scaduta o non è mai esistita. Determina il motivo per cui le chiavi non sono nella cache e sviluppa le strategie appropriate per averle nella cache.

[Risorse]:

- [Obbligatorio] Misura e raccogli le prestazioni della cache delle applicazioni insieme ai valori di latenza e utilizzo della CPU per capire se è necessario apportare modifiche al time-to-live o ad altri componenti dell'applicazione. ElastiCache fornisce una serie di metriche di CloudWatch per le latenze aggregate per ogni struttura di dati. Queste metriche per le latenze sono calcolate utilizzando le statistiche commandstats del comando INFO di ElastiCache per Redis e non includono i tempi della rete e di I/O, ma solo il tempo impiegato da ElastiCache per Redis per elaborare le operazioni.

[Risorse]:

- [Monitoring best practices with Amazon ElastiCache for Redis using Amazon CloudWatch](#)
- [Best practice] Scegli la strategia di memorizzazione nella cache giusta per le tue esigenze. Un basso rapporto di riscontri della cache è causato dal volume di mancati riscontri della cache. Se il carico di lavoro è progettato per avere un volume ridotto di mancati riscontri della cache (come la comunicazione in tempo reale), è consigliabile esaminare le strategie di memorizzazione

nella cache e applicare le risoluzioni più appropriate per il carico di lavoro, ad esempio la strumentazione delle query per misurare la memoria e le prestazioni. Le strategie efficaci utilizzate per implementare il popolamento e la gestione della cache dipende dai dati del client che desideri memorizzare nella cache e dai modelli di accesso a tali dati. Ad esempio, è improbabile che utilizzi in un'applicazione di streaming la stessa strategia per i suggerimenti personalizzati e per le notizie più interessanti.

[Risorse]:

- [Strategie di caching](#)
- [Caching Best Practices](#)
- [Performance at Scale with Amazon ElastiCache Whitepaper](#)

EP 4: in che modo il carico di lavoro ottimizza l'uso delle risorse e delle connessioni di rete?

Introduzione della domanda: ElastiCache per Redis e Memcached sono supportati da molti client di applicazioni e le implementazioni possono variare. È necessario comprendere la gestione della rete e delle connessioni in uso per analizzare il potenziale impatto sulle prestazioni.

Vantaggio della domanda: l'uso corretto delle risorse di rete può migliorare l'efficienza delle prestazioni del cluster. I seguenti suggerimenti possono ridurre le richieste di rete e migliorare la latenza e la velocità di trasmissione effettiva del cluster.

- [Obbligatorio] Gestisci in modo proattivo le connessioni al tuo cluster ElastiCache.
 - Il pool di connessioni dell'applicazione riduce la quantità di sovraccarico sul cluster creato dall'apertura e dalla chiusura delle connessioni. Monitora il comportamento delle connessioni in Amazon CloudWatch utilizzando `CurrConnections` e `NewConnections`.
 - Evita di perdere le connessioni client chiudendole correttamente, quando opportuno. Le strategie di gestione delle connessioni includono la corretta chiusura delle connessioni non in uso e l'impostazione del timeout delle connessioni.
 - Per i carichi di lavoro Memcached, esiste una quantità di memoria configurabile riservata alla gestione delle connessioni chiamate, `memcached_connections_overhead`.
- [Consigliato] Comprimi gli oggetti di grandi dimensioni per ridurre la memoria e migliorare la velocità di trasmissione effettiva di rete.
 - La compressione dei dati può ridurre la velocità di trasmissione effettiva di rete richiesta (Gbps), ma aumenta la quantità di lavoro dell'applicazione per comprimere e decomprimere i dati.

- La compressione riduce anche la quantità di memoria consumata dalle chiavi.
- In base alle esigenze della tua applicazione, considera i compromessi tra rapporto di compressione e velocità di compressione.
- [Risorse]:
 - [Global Datastore di Amazon ElastiCache per Redis](#)
 - [Parametri Memcached specifici](#)
 - [Amazon ElastiCache for Redis 5.0.3 migliora la gestione I/O per incrementare le prestazioni](#)
 - [Configure Amazon ElastiCache for Redis for higher availability](#)

EP 5: come si gestisce l'eliminazione e/o l'espulsione delle chiavi?

Introduzione della domanda: i carichi di lavoro hanno requisiti e comportamenti previsti diversi quando un nodo del cluster si avvicina ai limiti di consumo della memoria. Amazon ElastiCache per Redis prevede policy diverse per la gestione di queste situazioni.

Vantaggio della domanda: la corretta gestione della memoria disponibile e la comprensione delle policy di espulsione contribuiscono a garantire la consapevolezza del comportamento del cluster quando i limiti di memoria delle istanze vengono superati.

- [Obbligatorio] Strumenta l'accesso ai dati per valutare quale policy applicare. Identifica una policy per la memoria massima appropriata per controllare se e come vengono eseguite le espulsioni sul cluster.
 - L'espulsione si verifica quando viene consumata la memoria massima del cluster e c'è una policy che consente l'operazione. Il comportamento del cluster in questa situazione dipende dalla policy di espulsione specificata. Questa policy può essere gestita utilizzando `maxmemory-policy` nel gruppo di parametri del cluster ElastiCache per Redis.
 - La policy predefinita `volatile-lru` libera la memoria espellendo le chiavi con un tempo di scadenza impostato (valore TTL). Le policy per l'utilizzo meno frequente (LFU) e meno recente (LRU) rimuovono le chiavi in base all'utilizzo.
 - Per i carichi di lavoro Memcached, esiste una policy predefinita per LRU che controlla le espulsioni su ogni nodo. Il numero di espulsioni sul cluster Amazon ElastiCache può essere monitorato utilizzando la metrica delle espulsioni in Amazon CloudWatch.
- [Best practice] Standardizza il comportamento di eliminazione per controllare l'impatto sulle prestazioni del cluster ed evitare imprevisti colli di bottiglia delle prestazioni.

- Quando si rimuovono esplicitamente le chiavi dal cluster nei carichi di lavoro di ElastiCache per Redis, UNLINK è equivalente a DEL, rimuove le chiavi specificate. Tuttavia, il comando esegue il recupero della memoria effettiva in un thread diverso, quindi non si blocca, contrariamente a DEL. La rimozione effettiva avviene successivamente in modo asincrono.
- Per i carichi di lavoro di ElastiCache per Redis 6.x, il comportamento del comando DEL può essere modificato nel gruppo di parametri utilizzando il parametro `lazyfree-lazy-user-del`.
- [Risorse]:
 - [Configurazione dei parametri di motore con i gruppi di parametri](#)
 - [UNLINK](#)
 - [Gestione finanziaria del cloud con AWS](#)

EP 6: come si modellano e si usano i dati in ElastiCache?

Introduzione della domanda: ElastiCache dipende fortemente dall'applicazione per le strutture di dati e il modello di dati utilizzati, ma deve anche considerare l'archivio dati sottostante (se presente). Esamina le strutture di dati di ElastiCache per Redis disponibili e assicurati di utilizzare quelle più appropriate per le tue esigenze.

Vantaggio della domanda: la modellazione dei dati in ElastiCache ha diversi livelli, tra cui il caso d'uso dell'applicazione, i tipi di dati e le relazioni tra gli elementi di dati. Inoltre, ogni tipo di dati e comando di ElastiCache per Redis ha proprie caratteristiche prestazionali ben documentate.

- [Best practice] Una best practice consiste nel ridurre la sovrascrittura involontaria dei dati. Utilizza una convenzione di denominazione che riduca al minimo la sovrapposizione dei nomi delle chiavi. La convenzione di denominazione delle strutture di dati utilizza il metodo gerarchico `APPNAME:CONTEXT:ID`, ad esempio `ORDER-APP:CUSTOMER:123`.

[Risorse]:

- [Key naming](#)
- [Best practice] I comandi di ElastiCache per Redis hanno una complessità temporale definita dalla notazione Big O. La complessità temporale di un comando è la rappresentazione algoritmica/matematica del suo impatto. Quando si introduce un nuovo tipo di dati nell'applicazione, è necessario esaminare attentamente la complessità temporale dei relativi comandi. I comandi con la complessità temporale $O(1)$ sono costanti nel tempo e non dipendono dalla dimensione dell'input, mentre i comandi con la complessità temporale $O(N)$ sono lineari nel tempo e sono soggetti alla dimensione dell'input. A causa della progettazione a thread singolo di ElastiCache per

Redis, un elevato volume di operazioni ad alta complessità temporale comporta una riduzione delle prestazioni e potenziali timeout operativi.

[Risorse]:

- [Comandi](#)
- [Best practice] Usa le API per ottenere la visibilità della GUI nel modello di dati del cluster.

[Risorse]:

- [Redis Commander](#)
- [Redis Browser](#)
- [Redsmin](#)

EP 7: come si registrano i comandi a esecuzione lenta nel cluster Amazon ElastiCache?

Introduzione della domanda: vantaggi dell'ottimizzazione delle prestazioni attraverso l'acquisizione, l'aggregazione e la notifica di comandi di lunga durata. Comprendendo il tempo necessario per l'esecuzione dei comandi, è possibile determinare quali comandi comportano prestazioni scadenti e quali impediscono al motore di funzionare in modo ottimale. Amazon ElastiCache per Redis include anche la possibilità di inoltrare queste informazioni ad Amazon CloudWatch o Amazon Kinesis Data Firehose.

Vantaggio della domanda: l'accesso a una posizione permanente dedicata e l'invio di eventi di notifica per i comandi lenti possono aiutare a eseguire un'analisi dettagliata delle prestazioni e possono essere utilizzati per attivare eventi automatizzati.

- [Obbligatorio] Amazon ElastiCache per Redis esegue la versione 6.0 o successiva del motore, il gruppo di parametri è correttamente configurato e la registrazione SLOWLOG è abilitata sul cluster.
 - I parametri richiesti sono disponibili solo quando la compatibilità della versione del motore è impostata su Redis versione 6.0 o superiore.
 - La registrazione SLOWLOG si verifica quando l'esecuzione di un comando sul server richiede più tempo del valore specificato. Il comportamento del cluster dipende dai parametri `slowlog-log-slower-than` e `slowlog-max-len` del gruppo di parametri associato.
 - Le modifiche diventano effettive immediatamente.
- [Best practice] Utilizza le funzionalità di CloudWatch o Kinesis Data Firehose.

- Usa le funzionalità di filtro e allarme di CloudWatch, CloudWatch Logs Insights e Amazon Simple Notification Services per eseguire il monitoraggio delle prestazioni e la notifica degli eventi.
- Usa le funzionalità di streaming di Kinesis Data Firehose per archiviare i log SLOWLOG in uno spazio di archiviazione permanente o per attivare l'ottimizzazione automatica dei parametri del cluster.
- Determina se si adatta meglio alle tue esigenze il formato JSON o TEXT normale.
- Fornisci le autorizzazioni IAM per la pubblicazione su CloudWatch o Kinesis Data Firehose.
- [Consigliato] Configura `slowlog-log-slower-than` con un valore diverso da quello predefinito.
 - Questo parametro determina per quanto tempo un comando può essere eseguito nel motore Redis prima che venga registrato come comando a esecuzione lenta. Il valore predefinito è 10.000 microsecondi (10 millisecondi). Il valore predefinito potrebbe essere troppo alto per alcuni carichi di lavoro.
 - Determina il valore più appropriato per il tuo carico di lavoro in base alle esigenze delle applicazioni e ai risultati dei test, tuttavia considera che un valore troppo basso può generare un volume eccessivo di dati.
- [Consigliato] Lascia `slowlog-max-len` impostato sul valore predefinito.
 - Questo parametro determina il numero massimo di comandi a esecuzione lenta che possono essere acquisiti nella memoria Redis in un dato momento. Un valore pari a 0 disabilita l'acquisizione. Più alto è il valore, più elementi verranno archiviati in memoria, riducendo la possibilità che informazioni importanti vengano espulse prima di essere esaminate. Il valore predefinito è 128.
 - Il valore predefinito è appropriato per la maggior parte dei carichi di lavoro. Se è necessario analizzare i dati in una finestra temporale estesa dalla CLI di Redis tramite il comando SLOWLOG, valuta la possibilità di aumentare questo valore. In tal modo più comandi possono rimanere nella memoria Redis.

I dati SLOWLOG inviati a CloudWatch Logs o Kinesis Data Firehose vengono memorizzati in modo persistente e possono essere analizzati all'esterno del sistema ElastiCache, riducendo la necessità di archiviare un elevato numero di comandi a esecuzione lenta nella memoria Redis.

- [Risorse]:
 - [How do I turn on Redis Slow log in an ElastiCache for Redis cache cluster?](#)
 - [Consegna dei registri](#)
 - [Parametri specifici di Redis](#)
 - <https://aws.amazon.com/cloudwatch/> Amazon CloudWatch

- [Amazon Kinesis Data Firehose](#)

EP 8: in che modo il dimensionamento automatico aiuta ad aumentare le prestazioni del cluster ElastiCache?

Introduzione della domanda: implementando la funzionalità di dimensionamento automatico di Redis, i componenti ElastiCache possono adattarsi nel tempo per aumentare o diminuire automaticamente le partizioni o le repliche desiderate. Questa operazione può essere eseguita implementando il monitoraggio degli obiettivi o la policy di dimensionamento pianificata.

Vantaggio della domanda: la comprensione e la pianificazione dei picchi del carico di lavoro possono garantire prestazioni della cache migliorate e continuità aziendale. Il dimensionamento automatico di ElastiCache per Redis monitora continuamente l'utilizzo della memoria CPU per garantire che il cluster funzioni ai livelli di prestazioni desiderati.

- [Obbligatorio] Quando si avvia un cluster per ElastiCache per Redis:
 1. Assicurati che la modalità cluster sia abilitata
 2. Assicurati che l'istanza faccia parte della famiglia dei tipi e delle dimensioni che supportano il dimensionamento automatico
 3. Assicurati che il cluster non sia in esecuzione in datastore globali, Outpost o zone locali

[Risorse]:

- [Ridimensionamento dei cluster in Redis \(modalità cluster attivata\)](#)
- [Utilizzo di Auto Scaling con i shard](#)
- [Utilizzo di Auto Scaling con le repliche](#)
- [Best practice] Determina se il tuo carico di lavoro è pesante in lettura o in scrittura quando definisci la policy di dimensionamento. Per le prestazioni ottimali, utilizza una sola metrica di monitoraggio. Ti consigliamo di evitare l'uso di più policy per una dimensione perché le policy di dimensionamento automatico si dimensionano quando l'obiettivo viene raggiunto, ma si ridimensionano solo quando tutte le policy di monitoraggio degli obiettivi sono pronte per farlo.

[Risorse]:

- [Policy di Auto Scaling](#)
- [Definizione di una policy di dimensionamento](#)
- [Best practice] Il monitoraggio delle prestazioni nel tempo può aiutarti a rilevare i cambiamenti del carico di lavoro che rimarrebbero inosservati se monitorati solo in un determinato momento. Puoi

analizzare le metriche CloudWatch corrispondenti per l'utilizzo del cluster in un periodo di quattro settimane per determinare la soglia del valore dell'obiettivo. Se non sei ancora certo del valore da scegliere, ti consigliamo di iniziare con il valore predefinito minimo supportato della metrica.

[Risorse]:

- [Monitoraggio dell'utilizzo con i parametri di CloudWatch](#)
- [Consigliato] Consigliamo di testare l'applicazione con i carichi di lavoro minimi e massimi previsti, per identificare il numero esatto di partizioni/repliche necessarie al cluster per sviluppare le policy di dimensionamento e contenere i problemi di disponibilità.

[Risorse]:

- [Registrazione un target scalabile](#)
- [Registrazione un target scalabile](#)

Pilastro dell'ottimizzazione dei costi di Amazon ElastiCache Well-Architected Lens

Il pilastro dell'ottimizzazione dei costi si concentra sull'evitare costi inutili. Gli argomenti chiave includono la comprensione e il controllo delle spese, la selezione del tipo di nodo più appropriato (utilizza istanze che supportano la suddivisione dei dati su più livelli in base alle esigenze del carico di lavoro), il numero corretto dei tipi di risorse (repliche lette), l'analisi delle spese nel tempo e il dimensionamento per soddisfare le esigenze aziendali senza costi eccessivi.

Argomenti

- [COSTI 1: come si identificano e si monitorano i costi associati alle risorse ElastiCache? Come si sviluppano meccanismi per consentire agli utenti di creare, gestire ed eliminare le risorse create?](#)
- [COSTI 2: come si utilizzano gli strumenti di monitoraggio continuo per ottimizzare i costi associati alle risorse di ElastiCache?](#)
- [COSTI 3: è necessario utilizzare un tipo di istanza che supporti la suddivisione dei dati in più livelli? Quali sono i vantaggi di una suddivisione dei dati in più livelli? Quando non utilizzare le istanze di suddivisione dei dati in più livelli?](#)

COSTI 1: come si identificano e si monitorano i costi associati alle risorse ElastiCache? Come si sviluppano meccanismi per consentire agli utenti di creare, gestire ed eliminare le risorse create?

Introduzione della domanda: la comprensione delle metriche di costo richiede la partecipazione e la collaborazione di più team: ingegneria del software, gestione dei dati, proprietari dei prodotti, finanza e leadership. L'identificazione dei principali fattori di costo richiede che tutte le parti coinvolte comprendano le leve di controllo dell'utilizzo dei servizi e i compromessi di gestione dei costi che spesso è la differenza fondamentale tra le attività di ottimizzazione dei costi efficaci e meno efficaci. Assicurarti di disporre di processi e strumenti per tenere traccia delle risorse create durante lo sviluppo, la produzione e il ritiro e gestire i costi associati a ElastiCache.

Vantaggio della domanda: il monitoraggio continuo di tutti i costi associati al carico di lavoro richiede una conoscenza approfondita dell'architettura che include ElastiCache come componente. Inoltre, è necessario disporre di un piano di gestione dei costi per raccogliere e confrontare l'utilizzo rispetto al budget.

- [Obbligatorio] Istituisce un Centro di eccellenza del Cloud (CCoE) con una carta costitutiva per definire, tracciare e agire in base alle metriche relative all'utilizzo di ElastiCache da parte delle organizzazioni. Se un CCoE esiste e funziona, assicurati che possa leggere e tenere traccia dei costi associati a ElastiCache. Quando vengono create le risorse, utilizza i ruoli e le policy IAM per verificare che solo i team e i gruppi specifici possano creare istanze delle risorse. In tal modo i costi sono associati ai risultati aziendali e viene stabilita una chiara linea di responsabilità, dal punto di vista dei costi.
 1. Il CCoE deve identificare, definire e pubblicare metriche di costo aggiornate regolarmente, su base mensile, sull'utilizzo chiave di ElastiCache per i dati di categoria come:
 - a. Tipi di nodi utilizzati e relativi attributi: istanze standard e ottimizzate per la memoria, istanze on demand e riservate, regioni e zone di disponibilità
 - b. Tipi di ambienti: gratuito, sviluppo, test e produzione
 - c. Strategie di backup, archiviazione e conservazione
 - d. Trasferimento di dati all'interno e tra le regioni
 - e. Istanze in esecuzione su Amazon Outposts
 2. Il CCoE è composto da un team interfunzionale con rappresentanze non esclusive dei team di software engineering, gestione dei dati, team di prodotto, finanza e team dirigenziali dell'organizzazione.

[Risorse]:

- [Create a Cloud Center of Excellence](#)
 - [Prezzi di Amazon ElastiCache](#)
- [Obbligatorio] Utilizza i tag di allocazione dei costi per tenere traccia dei costi con un basso livello di granularità. Usa Gestione costi AWS per visualizzare, comprendere e gestire i costi e l'utilizzo di AWS nel tempo.
 1. Utilizza i tag per organizzare le tue risorse e tenere traccia dei costi di AWS a livello dettagliato. Dopo aver attivato i tag di allocazione dei costi, AWS li utilizza per organizzare i costi delle risorse nel proprio report di allocazione costi, per semplificare la categorizzazione e il monitoraggio dei costi di AWS. AWS offre due tipi di tag di allocazione dei costi, tag generati da AWS e tag definiti dall'utente. AWS definisce, crea e applica i tag generati da AWS per tuo conto, mentre tu definisci, crei e applichi i tag definiti dall'utente. È necessario attivare entrambi i tipi di tag separatamente per poterli visualizzare in Gestione dei costi o in un report di allocazione dei costi.
 2. Per organizzare le fatture AWS al fine di riflettere la struttura dei costi, puoi utilizzare i tag di allocazione dei costi. Quando aggiungi i tag di allocazione dei costi alle tue risorse in Amazon ElastiCache, puoi tenere traccia dei costi raggruppando le spese nelle fatture mediante i valori dei tag della risorsa. Puoi anche combinare i tag per monitorare i costi con un livello di dettagli maggiore.

[Risorse]:

- [Utilizzo dei tag per l'allocazione dei costi AWS](#)
 - [Monitoraggio dei costi con i tag di allocazione dei costi](#)
 - [AWS Cost Explorer](#)
- [Best practice] Collega i costi di ElastiCache alle metriche che riguardano l'intera organizzazione.
 1. Prendi in considerazione le metriche aziendali e le metriche operative come la latenza: quali concetti del tuo modello di business sono comprensibili in tutti i ruoli? Le metriche devono essere comprensibili per il maggior numero possibile di ruoli all'interno dell'organizzazione.
 2. Ad esempio, utenti serviti simultaneamente, latenza massima e media per operazione e utente, punteggi di coinvolgimento degli utenti, percentuali di reso degli utenti/settimana, durata della sessione/utente, tasso di abbandono, percentuale di riscontri nella cache e chiavi monitorate

[Risorse]:

- [Monitoraggio dell'utilizzo con i parametri di CloudWatch](#)

- [Positivo] Mantieni una visibilità operativa e dell'architettura aggiornata su metriche e costi dell'intero carico di lavoro che utilizza ElastiCache.
 1. Comprendendo l'intero ecosistema di soluzioni, ElastiCache tende a far parte di un ecosistema completo di servizi AWS nel set tecnologico, da client a gateway API, Redshift e QuickSight per gli strumenti di reporting (ad esempio).
 2. Mappa i componenti della soluzione (client, connessioni, sicurezza, operazioni in memoria, archiviazione, automazione delle risorse, accesso e gestione dei dati) nel tuo diagramma di architettura. Ogni livello si collega all'intera soluzione e ha proprie esigenze e funzionalità che aumentano e/o aiutano a gestire il costo complessivo.
 3. Il diagramma deve includere l'uso di calcolo, rete, archiviazione, policy del ciclo di vita, raccolta di metriche e gli elementi operativi e funzionali di ElastiCache dell'applicazione
 4. È probabile che i requisiti del carico di lavoro si evolvano nel tempo ed è essenziale continuare a mantenere e documentare la comprensione dei componenti sottostanti e dei propri obiettivi funzionali primari per rimanere proattivi nella gestione dei costi del carico di lavoro.
 5. Il supporto esecutivo per visibilità, responsabilità, priorità e risorse è fondamentale per disporre di una strategia di gestione dei costi efficace per ElastiCache.

COSTI 2: come si utilizzano gli strumenti di monitoraggio continuo per ottimizzare i costi associati alle risorse di ElastiCache?

Introduzione della domanda: è necessario puntare a un giusto equilibrio tra i costi di ElastiCache e le metriche delle prestazioni delle applicazioni. Amazon CloudWatch offre visibilità sui principali parametri operativi che possono aiutarti a valutare se le risorse di ElastiCache sono sovrautilizzate o meno, rispetto alle tue esigenze. Dal punto di vista dell'ottimizzazione dei costi, è necessario comprendere quando il provisioning è eccessivo ed essere in grado di sviluppare meccanismi appropriati per ridimensionare le risorse di ElastiCache rispettando al contempo le esigenze operative, di disponibilità, resilienza e prestazioni.

Vantaggio della domanda: in un ambiente ideale, hai messo a disposizione risorse sufficienti per soddisfare le esigenze operative del tuo carico di lavoro e non disponi di risorse sottoutilizzate che possono portare costi non ottimali. È necessario essere in grado di identificare ed evitare di utilizzare risorse di ElastiCache sovradimensionate per lunghi periodi di tempo.

- [Obbligatorio] Usa CloudWatch per monitorare i tuoi cluster ElastiCache e analizzare il modo in cui queste metriche si riferiscono alle dashboard di Esploratore dei costi AWS

1. ElastiCache fornisce parametri a livello di host (ad esempio, utilizzo della CPU) e parametri che sono specifici del software del motore di cache (ad esempio, cache gets e cache misses). Questi parametri vengono misurati e pubblicati per ogni nodo di cache in intervalli di 60 secondi.
2. Le metriche delle prestazioni di ElastiCache (CPUUtilization, EngineUtilization, SwapUsage, CurrConnections ed Evictions) possono indicare che è necessario il dimensionamento verticale (utilizza tipi di nodi di cache più grandi/più piccoli) o il dimensionamento orizzontale (aggiungi più/meno partizioni). Comprendi le implicazioni in termini di costi delle decisioni di dimensionamento creando una matrice di playbook che stimi i costi aggiuntivi e i tempi minimi e massimi necessari per raggiungere le soglie di prestazioni delle applicazioni.

[Risorse]:

- [Monitoraggio dell'utilizzo con i parametri di CloudWatch](#)
 - [Quali parametri è opportuno monitorare?](#)
 - [Prezzi di Amazon ElastiCache](#)
- [Obbligatorio] Comprendi e documenta la tua strategia di backup e le implicazioni in termini di costi.
 1. Con ElastiCache, i backup vengono archiviati in Amazon S3, che fornisce uno spazio di archiviazione durevole. È necessario comprendere le implicazioni in termini di costi in relazione alla capacità di ripristino in caso di guasti.
 2. Abilita i backup automatici che eliminano i file di backup che hanno superato il limite di conservazione.

[Risorse]:

- [Pianificazione di backup automatici](#)
 - [Prezzi di Amazon S3](#)
- [Best practice] Usa i nodi riservati per le tue istanze come strategia deliberata per gestire i costi dei carichi di lavoro che sono ben appresi e documentati. I nodi riservati vengono addebitati un costo anticipato che dipende dal tipo di nodo e dalla durata della prenotazione, ovvero uno o tre anni. Tale addebito è molto inferiore all'addebito orario di utilizzo previsto in caso di nodi on demand.
 1. Potrebbe essere necessario gestire i cluster ElastiCache utilizzando nodi on demand fino a quando non avrai raccolto dati sufficienti per stimare i requisiti delle istanze riservate. Pianifica e documenta le risorse necessarie per soddisfare le tue esigenze e confronta i costi previsti tra i diversi tipi di istanza (on demand o riservata)
 2. Valuta regolarmente i nuovi tipi di nodi di cache disponibili e se ha senso, dal punto di vista dei costi e delle metriche operative, migrare il parco istanze a nuovi tipi di nodi di cache

COSTI 3: è necessario utilizzare un tipo di istanza che supporti la suddivisione dei dati in più livelli? Quali sono i vantaggi di una suddivisione dei dati in più livelli? Quando non utilizzare le istanze di suddivisione dei dati in più livelli?

Introduzione della domanda: la selezione del tipo di istanza appropriato può avere non solo un impatto sulle prestazioni e sul livello di servizio, ma anche un impatto finanziario. I tipi di istanza hanno costi diversi. La selezione di uno o più tipi di istanze di grandi dimensioni in grado di soddisfare tutte le esigenze di archiviazione in memoria potrebbe essere una decisione naturale. Tuttavia, potrebbe avere un impatto significativo sui costi man mano che il progetto matura. Per garantire la selezione del tipo di istanza corretto è necessario un esame periodico del tempo di inattività degli oggetti ElastiCache.

Vantaggio della domanda: una chiara comprensione dell'impatto dei vari tipi di istanze sui costi attuali e futuri. Le modifiche marginali o periodiche del carico di lavoro non devono causare variazioni sproporzionate dei costi. Se il carico di lavoro lo consente, i tipi di istanze che supportano la suddivisione dei dati in più livelli offrono un prezzo migliore per l'archiviazione disponibile. Grazie allo spazio di archiviazione SSD disponibile per istanza, le istanze con suddivisione dei dati in più livelli supportano una capacità totale di dati per istanza molto più elevata.

- [Obbligatorio] Comprendi i limiti delle istanze con suddivisione dei dati in più livelli
 1. Disponibile solo per i cluster ElastiCache per Redis.
 2. Solo tipi di istanza limitati supportano la suddivisione dei dati in più livelli.
 3. Solo ElastiCache per Redis versione 6.2 e successive è supportato.
 4. Gli oggetti di grandi dimensioni non vengono sostituiti in SSD. Gli oggetti superiori a 128 MiB vengono conservati in memoria.

[Risorse]:

- [Tiering di dati](#)
- [Prezzi di Amazon ElastiCache](#)
- [Obbligatorio] Scopri la percentuale del database che viene regolarmente utilizzata dal carico di lavoro.
 1. Le istanze con suddivisione dei dati in più livelli sono ideali per i carichi di lavoro che spesso accedono a una piccola parte del set di dati complessivo ma richiedono comunque un accesso rapido ai dati rimanenti. In altre parole, il rapporto tra dati da "ad accesso frequente" ad "a caldo" è di circa 20:80.
 2. Sviluppa il monitoraggio a livello di cluster dei tempi di inattività degli oggetti.

3. Le grandi implementazioni di oltre 500 GB di dati sono buone candidate
- [Obbligatorio] Tieni presente che le istanze con suddivisione dei dati in più livelli non sono opzionali per determinati carichi di lavoro.
 1. L'accesso agli oggetti usati meno frequentemente comporta un costo in termini di prestazioni ridotto, poiché questi vengono sostituiti nel SSD locale. Se la tua applicazione è sensibile ai tempi di risposta, verifica l'impatto sul carico di lavoro.
 2. Non adatto per cache che memorizzano principalmente oggetti di grandi dimensioni superiori a 128 MiB.

[Risorse]:

- [Limitazioni](#)
- [Best practice] I tipi di istanze riservate supportano la suddivisione dei dati in più livelli. Ciò garantisce il costo più basso in termini di quantità di archiviazione di dati per istanza.
 1. Potrebbe essere necessario gestire i cluster ElastiCache utilizzando istanze con suddivisione dei dati in più livelli non basate sui dati fino a quando non si ha una migliore comprensione dei requisiti.
 2. Analizza il modello di utilizzo dei dati dei cluster ElastiCache.
 3. Crea un processo automatico che raccolga periodicamente i tempi di inattività degli oggetti.
 4. Se noti che un'alta percentuale (circa l'80%) di oggetti è inattiva per un periodo di tempo ritenuto appropriato per il tuo carico di lavoro, documenta i risultati e suggerisci di migrare il cluster su istanze che supportano la suddivisione dei dati in più livelli.
 5. Valuta regolarmente i nuovi tipi di nodi di cache disponibili e se ha senso, dal punto di vista dei costi e delle metriche operative, migrare il parco istanze a nuovi tipi di nodi di cache.

[Risorse]:

- [OBJECT IDLETIME](#)
- [Prezzi di Amazon ElastiCache](#)

Procedure e best practice comuni per la risoluzione dei problemi

Argomenti

- [Problemi di connessione](#)
- [Errori del client Redis](#)

- [Risoluzione dei problemi di latenza elevata in Serverless ElastiCache](#)
- [Risoluzione dei problemi di limitazione in Serverless ElastiCache](#)
- [Argomenti correlati](#)

Problemi di connessione

Se non riesci a connetterti alla ElastiCache cache, prendi in considerazione una delle seguenti opzioni:

1. Utilizzo di TLS: se si verifica un blocco della connessione durante il tentativo di connessione all'ElastiCache endpoint, è possibile che non si stia utilizzando TLS nel client. Se utilizzi ElastiCache Serverless, la crittografia in transito è sempre abilitata. Assicurati che il tuo client utilizzi TLS per connettersi alla cache. [Scopri di più sulla connessione a una cache abilitata per TLS qui.](#)
2. VPC: le ElastiCache cache sono accessibili solo dall'interno di un VPC. Assicurati che l'istanza EC2 da cui accedi alla cache e la ElastiCache cache siano create nello stesso VPC. In alternativa, devi abilitare il [peering VPC tra](#) il VPC in cui risiede l'istanza EC2 e il VPC in cui stai creando la cache.
3. Gruppi di sicurezza: ElastiCache utilizza i gruppi di sicurezza per controllare l'accesso alla cache. Considera i seguenti aspetti:
 - a. Assicurati che il gruppo di sicurezza utilizzato dalla ElastiCache cache consenta l'accesso in entrata dalla tua istanza EC2. Leggi [qui](#) per scoprire come configurare correttamente le regole in entrata nel tuo gruppo di sicurezza.
 - b. Assicurati che il gruppo di sicurezza utilizzato dalla ElastiCache cache consenta l'accesso alle porte della cache (6379 e 6380 per le versioni serverless e 6379 per impostazione predefinita per quelle progettate autonomamente). ElastiCache utilizza queste porte per accettare i comandi Redis. Scopri di più su come configurare l'accesso alle porte [qui](#).

Errori del client Redis

ElastiCache Serverless è accessibile solo tramite client Redis che supportano il protocollo Redis in modalità cluster. È possibile accedere ai cluster progettati autonomamente dai client Redis in entrambe le modalità, a seconda della configurazione del cluster.

Se riscontri errori Redis nel tuo client, considera quanto segue:

1. Modalità cluster: se si verificano errori CROSSLOT o errori con il comando [SELECT](#) Redis, è possibile che si stia tentando di accedere a una cache Cluster Mode Enabled con un client Redis che non supporta il protocollo Redis Cluster. ElastiCache Serverless supporta solo client Redis che supportano il protocollo cluster Redis. Se desideri utilizzare Redis in «Cluster Mode Disabled» (CMD), devi progettare il tuo cluster.
2. Errori CROSSLOT: se si verifica l'ERR CROSSLOT Keys in request don't hash to the same slot errore, è possibile che si stia tentando di accedere a chiavi che non appartengono allo stesso slot in una cache in modalità Cluster. Come promemoria, ElastiCache Serverless funziona sempre in modalità Cluster. Le operazioni a più chiavi, le transazioni o gli script Lua che coinvolgono più chiavi sono consentite solo se tutte le chiavi coinvolte si trovano nello stesso slot di hash.

[Per ulteriori best practice sulla configurazione dei client Redis, consulta questo post sul blog.](#)

Risoluzione dei problemi di latenza elevata in Serverless ElastiCache

Se il tuo carico di lavoro sembra presentare un'elevata latenza, puoi analizzare le `SuccessfulWriteRequestLatency` e `SuccessfulReadRequestLatency` metriche CloudWatch per verificare se la latenza è correlata a Serverless. Queste metriche misurano la latenza interna a ElastiCache Serverless: la latenza lato client e i tempi di viaggio di rete tra il client e l'endpoint Serverless non sono inclusi. ElastiCache

Alcune variabilità e i picchi occasionali non dovrebbero essere motivo di preoccupazione. Tuttavia, se la `Average` statistica mostra un forte aumento e persiste, dovresti controllare la `Personal Health Dashboard` AWS Health Dashboard e la tua `Personal Health Dashboard` per ulteriori informazioni. Se necessario, valuta la possibilità di aprire una richiesta di supporto con `AWS Support`.

Prendi in considerazione le seguenti best practice e strategie per ridurre la latenza:

- **Abilita Read from Replica:** se la tua applicazione lo consente, ti consigliamo di abilitare la funzionalità «Read from Replica» nel tuo client Redis per scalare le letture e ottenere una latenza inferiore. Se abilitata, ElastiCache Serverless tenta di indirizzare le richieste di lettura verso i nodi di cache di replica che si trovano nella stessa zona di disponibilità (AZ) del client, evitando così la latenza di rete Cross-AZ. Tieni presente che l'attivazione della funzionalità Read from Replica nel client significa che l'applicazione accetta l'eventuale coerenza dei dati. L'applicazione potrebbe ricevere dati più vecchi per qualche tempo se si tenta di leggere dopo aver scritto su una chiave.
- **Assicurati che l'applicazione sia distribuita nelle stesse AZ della cache:** potresti osservare una latenza lato client più elevata se l'applicazione non è distribuita negli stessi AZ della cache. Quando

si crea una cache serverless, è possibile fornire le sottoreti da cui l'applicazione accederà alla cache e ElastiCache Serverless crea endpoint VPC in tali sottoreti. Assicurati che l'applicazione sia distribuita nelle stesse AZ. In caso contrario, l'applicazione potrebbe subire un hop Cross-AZ quando accede alla cache, con conseguente maggiore latenza lato client.

- Riutilizza le connessioni: le richieste ElastiCache serverless vengono effettuate tramite una connessione TCP abilitata per TLS utilizzando il protocollo RESP. L'avvio della connessione (inclusa l'autenticazione della connessione, se configurata) richiede tempo, quindi la latenza della prima richiesta è superiore a quella tipica. Le richieste su una connessione già inizializzata offrono ElastiCache una latenza costantemente bassa. Per questo motivo, dovresti prendere in considerazione l'utilizzo del pool di connessioni o il riutilizzo delle connessioni Redis esistenti.
- Velocità di scalabilità: ElastiCache Serverless si ridimensiona automaticamente all'aumentare della frequenza delle richieste. Un aumento improvviso e significativo della frequenza di richieste, superiore alla velocità di scalabilità di ElastiCache Serverless, può comportare una latenza elevata per qualche tempo. ElastiCache In genere, Serverless può aumentare rapidamente la frequenza di richieste supportata, impiegando fino a 10-12 minuti per raddoppiare la frequenza delle richieste.
- Ispeziona i comandi a esecuzione prolungata: alcuni comandi Redis, inclusi gli script Lua o i comandi su strutture di dati di grandi dimensioni, possono essere eseguiti a lungo. Per identificare questi comandi, ElastiCache pubblica metriche a livello di comando. Con [ElastiCache Serverless](#) puoi usare le metriche. `BasedECPUs`
- Richieste limitate: quando le richieste vengono limitate in ElastiCache Serverless, è possibile che si verifichi un aumento della latenza lato client nell'applicazione. [Quando le richieste vengono limitate in ElastiCache Serverless, dovresti notare un aumento della metrica Serverless. `ThrottledRequests ElastiCache`](#) Consulta la sezione seguente per la risoluzione dei problemi relativi alle richieste limitate.
- Distribuzione uniforme di chiavi e richieste: in ElastiCache Redis, una distribuzione non uniforme di chiavi o richieste per slot può comportare un hot slot che può comportare una latenza elevata. ElastiCache Serverless supporta fino a 30.000 ECPU al secondo (90.000 ECPU al secondo quando si utilizza Read from Replica) su un singolo slot, in un carico di lavoro che esegue semplici comandi SET/GET. Ti consigliamo di valutare la distribuzione delle chiavi e delle richieste tra gli slot e di garantire una distribuzione uniforme se la frequenza delle richieste supera questo limite.

Risoluzione dei problemi di limitazione in Serverless ElastiCache

Nelle architetture orientate ai servizi e nei sistemi distribuiti, la limitazione della velocità con cui le chiamate API vengono elaborate dai vari componenti del servizio si chiama limitazione (della

larghezza di banda della rete). Ciò attenua i picchi, controlla le discrepanze nella velocità di trasmissione dei componenti e consente ripristini più prevedibili in caso di eventi operativi imprevisti. ElastiCache Serverless è progettato per questi tipi di architetture e la maggior parte dei client Redis dispone di nuovi tentativi integrati per le richieste limitate. Un certo grado di limitazione (della larghezza di banda della rete) non è necessariamente un problema per l'applicazione, ma una limitazione (della larghezza di banda della rete) persistente di una parte sensibile alla latenza del flusso di lavoro dei dati può influire negativamente sull'esperienza dell'utente e ridurre l'efficienza complessiva del sistema.

[Quando le richieste vengono limitate in Serverless, dovresti notare un aumento della ElastiCache metrica Serverless. ThrottledRequests ElastiCache](#) Se notate un numero elevato di richieste limitate, considerate quanto segue:

- **Velocità di scalabilità:** ElastiCache Serverless si ridimensiona automaticamente man mano che si acquisiscono più dati o si aumenta la frequenza delle richieste. Se l'applicazione si adatta più velocemente della scalabilità Serverless, le richieste potrebbero essere limitate mentre ElastiCache Serverless si ridimensiona per adattarsi al carico di lavoro. ElastiCache ElastiCache In genere, la modalità serverless consente di aumentare rapidamente le dimensioni di storage, impiegando fino a 10-12 minuti per raddoppiare le dimensioni di archiviazione nella cache.
- **Distribuzione uniforme di chiavi e richieste:** in ElastiCache Redis, una distribuzione non uniforme di chiavi o richieste per slot può causare un hot slot. Un hot slot può comportare una limitazione delle richieste se la frequenza delle richieste verso un singolo slot supera le 30.000 ECPU al secondo, in un carico di lavoro che esegue semplici comandi SET/GET.
- **Leggi dalla replica:** se l'applicazione lo consente, valuta la possibilità di utilizzare la funzione «Leggi dalla replica». La maggior parte dei client Redis può essere configurata per «scalare le letture» per indirizzare le letture ai nodi di replica. Questa funzionalità consente di scalare il traffico di lettura. Inoltre, ElastiCache Serverless indirizza automaticamente la lettura dalle richieste di replica ai nodi nella stessa zona di disponibilità dell'applicazione, con conseguente riduzione della latenza. Quando Read from Replica è abilitato, è possibile ottenere fino a 90.000 ECPU al secondo su un singolo slot, per carichi di lavoro con semplici comandi SET/GET.

Argomenti correlati

- [Ulteriori passaggi per la risoluzione dei problemi](#)
- [the section called “Best practice e strategie di caching”](#)

Ulteriori passaggi per la risoluzione dei problemi

I seguenti elementi devono essere verificati durante la risoluzione dei problemi di connettività persistenti con ElastiCache:

Argomenti

- [Gruppi di sicurezza](#)
- [Liste di controllo accessi \(ACL\) di rete](#)
- [Tabelle di instradamento](#)
- [Risoluzione DNS](#)
- [Identificazione dei problemi con la diagnostica lato server](#)
- [Convalida della connettività di rete](#)
- [Limiti relativi alla rete](#)
- [Utilizzo CPU](#)
- [Connessioni terminate dal lato server](#)
- [Risoluzione dei problemi lato client per istanze Amazon EC2](#)
- [Dissezione del tempo necessario per completare una singola richiesta](#)

Gruppi di sicurezza

I gruppi di sicurezza sono firewall virtuali che proteggono il ElastiCache client (istanza EC2, AWS Lambda funzione, contenitore Amazon ECS, ecc.) e la cache. ElastiCache I gruppi di sicurezza sono dotati di stato, il che significa che dopo che il traffico in entrata o in uscita è consentito, le risposte per tale traffico verranno automaticamente autorizzate nel contesto di tale gruppo di sicurezza specifico.

La funzionalità con stato richiede che il gruppo di sicurezza monitori tutte le connessioni autorizzate ed esiste un limite per le connessioni monitorate. Se il limite viene raggiunto, le nuove connessioni avranno esito negativo. Consulta la sezione dedicata alla risoluzione dei problemi per sapere se i limiti sono stati raggiunti dal client o dal lato. ElastiCache

È possibile assegnare contemporaneamente un singolo gruppo di sicurezza al client e al ElastiCache cluster o singoli gruppi di sicurezza per ciascuno.

In entrambi i casi, è necessario consentire il traffico TCP in uscita sulla ElastiCache porta dall'origine e il traffico in entrata sulla stessa porta verso. ElastiCache La porta predefinita è 11211 per

Memcached e 6379 per Redis. Per impostazione predefinita, i gruppi di sicurezza autorizzano tutto il traffico in uscita. In questo caso, è necessaria solo la regola in entrata nel gruppo di sicurezza di destinazione.

Per ulteriori informazioni, consulta [Modelli di accesso per l'accesso a un ElastiCache cluster in un Amazon VPC](#).

Liste di controllo accessi (ACL) di rete

Liste di controllo accessi di rete (ACL) sono regole senza stato. Il traffico deve essere consentito in entrambe le direzioni (in entrata e in uscita) per avere esito positivo. Le ACL di rete vengono assegnate alle sottoreti, non a risorse specifiche. È possibile assegnare lo stesso ACL ElastiCache e la stessa risorsa client, specialmente se si trovano nella stessa sottorete.

Per impostazione predefinita, le ACL di rete consentono tutti i traffici. Tuttavia, è possibile personalizzarle per negare o consentire il traffico. Inoltre, la valutazione delle regole delle ACL è sequenziale, ossia la regola con il numero più basso corrispondente al traffico consentirà o negherà tale traffico. La configurazione minima per consentire il traffico Redis è:

ACL di rete lato client:

- Regole in entrata:
- Numero della regola: preferibilmente inferiore a qualsiasi regola di diniego;
- Tipo: Regola TCP personalizzata;
- Protocollo: TCP
- Intervallo porte: 1024-65535
- Fonte: 0.0.0.0/0 (o crea regole individuali per le sottoreti del cluster) ElastiCache
- Consenti/Nega: Consenti

- Regole in uscita:
- Numero della regola: preferibilmente inferiore a qualsiasi regola di diniego;
- Tipo: Regola TCP personalizzata;
- Protocollo: TCP
- Intervallo porte: 6379
- Fonte: 0.0.0.0/0 (o le sottoreti del cluster. ElastiCache Tieni presente che l'utilizzo di IP specifici può creare problemi in caso di failover o scalabilità del cluster)

- Consenti/Nega: Consenti

ElastiCache ACL di rete:

- Regole in entrata:
 - Numero della regola: preferibilmente inferiore a qualsiasi regola di diniego;
 - Tipo: Regola TCP personalizzata;
 - Protocollo: TCP
 - Intervallo porte: 6379
 - Fonte: 0.0.0.0/0 (o crea regole individuali per le sottoreti del cluster) ElastiCache
 - Consenti/Nega: Consenti
- Regole in uscita:
 - Numero della regola: preferibilmente inferiore a qualsiasi regola di diniego;
 - Tipo: Regola TCP personalizzata;
 - Protocollo: TCP
 - Intervallo porte: 1024-65535
 - Fonte: 0.0.0.0/0 (o le sottoreti del cluster. ElastiCache Tieni presente che l'utilizzo di IP specifici può creare problemi in caso di failover o scalabilità del cluster)
 - Consenti/Nega: Consenti

Per ulteriori informazioni, consultare [ACL di rete](#).

Tabelle di instradamento

Analogamente alle ACL di rete, ogni sottorete può avere tabelle di routing diverse. Se i client e il ElastiCache cluster si trovano in sottoreti diverse, assicurati che le tabelle di routing consentano loro di raggiungersi.

Gli ambienti più complessi, che coinvolgono più routing dinamici di VPC, o firewall di rete, possono risultare difficili da risolvere. Consultare [Convalida della connettività di rete](#) per verificare che le impostazioni di rete siano appropriate.

Risoluzione DNS

ElastiCache fornisce gli endpoint del servizio in base ai nomi DNS. Gli endpoint disponibili sono `Configuration`, `Primary`, `Reader` e `Node`. Per maggiori informazioni, consultare [Trovare gli endpoint di connessione](#).

In caso di failover o di modifica del cluster, l'indirizzo associato al nome dell'endpoint potrebbe cambiare e verrà aggiornato automaticamente.

Le impostazioni DNS personalizzate (ad esempio, che non utilizzano il servizio DNS VPC) potrebbero non essere a conoscenza dei nomi DNS forniti di ElastiCache. Assicurati che il tuo sistema sia in grado di risolvere correttamente gli endpoint di ElastiCache utilizzando strumenti di sistema come `dig` (come mostrato di seguito) o `nslookup`.

```
$ dig +short example.xxxxxx.ng.0001.use1.cache.amazonaws.com
example-001.xxxxxx.0001.use1.cache.amazonaws.com.
1.2.3.4
```

È inoltre possibile forzare la risoluzione dei nomi tramite il servizio del DNS VPC:

```
$ dig +short example.xxxxxx.ng.0001.use1.cache.amazonaws.com @169.254.169.253
example-001.tihewd.0001.use1.cache.amazonaws.com.
1.2.3.4
```

Identificazione dei problemi con la diagnostica lato server

CloudWatch le metriche e le informazioni sul tempo di funzionamento del motore di ElastiCache sono fonti o informazioni comuni per identificare potenziali fonti di problemi di connessione. Una buona analisi inizia comunemente con i seguenti elementi:

- **Utilizzo della CPU:** Redis è un'applicazione multithread. Tuttavia, l'esecuzione di ogni comando avviene in un singolo thread (principale). Per questo motivo, ElastiCache fornisce le metriche `CPUUtilization` e `EngineCPUUtilization`. `EngineCPUUtilization` fornisce l'utilizzo della CPU dedicato al processo Redis e `CPUUtilization` l'utilizzo su tutte le vCPU. I nodi con più di una vCPU di solito hanno valori diversi per `CPUUtilization` e `EngineCPUUtilization`, il secondo è comunemente più alto. Un elevato `EngineCPUUtilization` può essere causato da un numero elevato di richieste o operazioni complesse che richiedono una notevole quantità di tempo di CPU per il completamento. È possibile identificare entrambi con i seguenti elementi:

- Numero elevato di richieste: verifica la presenza di aumenti su altri parametri che corrispondono al modello `EngineCPUUtilization`. I parametri utili sono:
 - `CacheHits` e `CacheMisses`: il numero di richieste corrette o richieste che non hanno trovato un elemento valido nella cache. Se il rapporto tra perdite e successi è elevato, l'applicazione sta sprecando tempo e risorse con richieste infruttuose.
 - `SetTypeCmds` e `GetTypeCmds`: questi parametri correlati con `EngineCPUUtilization` possono aiutare a capire se il carico è significativamente più alto per le richieste di scrittura, misurato da `SetTypeCmds`, o letture, misurate da `GetTypeCmds`. Se il carico è prevalentemente di lettura, l'utilizzo di più repliche di lettura può bilanciare le richieste su più nodi e risparmiare il primario per le scritture. Nei cluster disattivati in modalità cluster, è possibile utilizzare le repliche di lettura creando una configurazione di connessione aggiuntiva nell'applicazione utilizzando l'endpoint di lettura. ElastiCache Per maggiori informazioni, consultare [Trovare gli endpoint di connessione](#). Le operazioni di lettura devono essere inviate a questa connessione aggiuntiva. Le operazioni di scrittura verranno eseguite attraverso l'endpoint primario regolare. In modalità cluster abilitata, è consigliabile utilizzare una libreria che supporti le repliche di lettura in modalità nativa. Con i flag giusti, la libreria sarà in grado di individuare automaticamente la topologia del cluster, i nodi di replica, abilitare le operazioni di lettura tramite il comando di Redis `READONLY` e inviare le richieste di lettura alle repliche.
- Elevato numero di connessioni:
 - `CurrConnections` e `NewConnections`: `CurrConnection` è il numero di connessioni stabilite al momento della raccolta dei datapoint, mentre `NewConnections` mostra quante connessioni sono state create nel periodo.

La creazione e la gestione delle connessioni implica un notevole overhead della CPU. Inoltre, l'handshake a tre vie TCP richiesto per creare nuove connessioni influirà negativamente sui tempi di risposta complessivi.

Un ElastiCache nodo con migliaia di dati `NewConnections` al minuto indica che una connessione viene creata e utilizzata con pochi comandi, il che non è ottimale. Mantenere le connessioni stabilite e riutilizzarle per nuove operazioni è una best practice. Ciò è possibile quando l'applicazione client supporta e implementa correttamente il pool di connessioni o le connessioni persistenti. Con il pool di connessioni, il numero di `currConnections` non ha grandi variazioni, e le `NewConnections` dovrebbero essere il più basso possibile. Redis fornisce prestazioni ottimali con un numero ridotto di `currConnections`. Mantenere `currConnection` nell'ordine di decine o centinaia riduce al minimo l'utilizzo di risorse per supportare singole connessioni come buffer client e cicli CPU per servire la connessione.

- **Velocità effettiva di rete**
 - Determina la larghezza di banda: ElastiCache i nodi hanno una larghezza di banda di rete proporzionale alla dimensione del nodo. Poiché le applicazioni hanno caratteristiche diverse, i risultati possono variare in base al carico di lavoro. Ad esempio, le applicazioni con un elevato tasso di richieste di piccole dimensioni tendono a influire di più sull'utilizzo della CPU rispetto alla velocità effettiva di rete, mentre le chiavi più grandi causeranno un maggiore utilizzo della rete. Per questo motivo, è consigliabile testare i nodi con il carico di lavoro effettivo per una migliore comprensione dei limiti.

La simulazione del carico dall'applicazione fornirebbe risultati più accurati. Tuttavia, gli strumenti relativi al valore di riferimento possono dare una buona idea dei limiti.

- Nei casi in cui le richieste sono prevalentemente letture, l'utilizzo delle repliche per le operazioni di lettura riduce il carico sul nodo primario. Se il caso d'uso è prevalentemente scritture, l'uso di molte repliche amplificherà l'utilizzo della rete. Per ogni byte scritto nel nodo primario, N byte verranno inviati alle repliche, essendo N il numero di repliche. La best practice per carichi di lavoro ad alta intensità di scrittura è quella di utilizzare Redis con ElastiCache la modalità cluster abilitata, in modo che le scritture possano essere bilanciate su più shard o scalare fino a un tipo di nodo con maggiori funzionalità di rete.
- I CloudWatchmetrics `NetworkBytesIn` e `NetworkBytesOut` forniscono rispettivamente la quantità di dati in entrata o in uscita dal nodo. `ReplicationBytes` è il traffico dedicato alla replica dei dati.

Per ulteriori informazioni, consulta [Limiti relativi alla rete](#).

- **Comandi complessi:** i comandi di Redis sono serviti su un singolo thread, ossia le richieste vengono servite in sequenza. Un singolo comando lento può influire su altre richieste e connessioni, culminando in time-out. L'uso di comandi che agiscono su più valori, chiavi o tipi di dati deve essere fatto con attenzione. Le connessioni possono essere bloccate o terminate a seconda del numero di parametri o delle dimensioni dei valori di input o output.

Un esempio famigerato è il comando KEYS. Esso spazza l'intero keyspace alla ricerca di un dato modello e blocca l'esecuzione di altri comandi durante la sua esecuzione. Redis utilizza la notazione "Big O" per descrivere la complessità dei suoi comandi.

Il comando delle chiavi ha complessità di tempo $O(N)$, dove N è il numero di chiavi presenti nel database. Pertanto, maggiore è il numero di chiavi, più lento sarà il comando. KEYS può causare problemi in diversi modi: se non viene utilizzato alcun modello di ricerca, il comando restituirà

tutti i nomi delle chiavi disponibili. Nei database con migliaia o milioni di elementi, verrà creato un output enorme e inonderà i buffer di rete.

Se viene utilizzato un modello di ricerca, solo le chiavi corrispondenti al modello torneranno al client. Tuttavia, il motore continua a spazzare l'intero keyspace alla sua ricerca e il tempo necessario per completare il comando sarà lo stesso.

Un'alternativa per KEYS è il comando SCAN. Si ripete sul keyspace e limita le iterazioni in un numero specifico di elementi, evitando blocchi prolungati sul motore.

La scansione ha il parametro COUNT, utilizzato per impostare la dimensione dei blocchi di iterazione. Il valore di default è 10 (10 elementi per iterazione).

A seconda del numero di elementi nel database, piccoli blocchi di valori COUNT richiedono più iterazioni per completare una scansione completa e valori più grandi manterranno il motore occupato più a lungo ad ogni iterazione. Mentre piccoli valori di conteggio renderanno SCAN più lento su database di grandi dimensioni, valori più grandi possono causare gli stessi problemi menzionati per KEYS.

Ad esempio, l'esecuzione del comando SCAN con valore di conteggio di 10 richiederà 100.000 ripetizioni su un database con 1 milione di chiavi. Se il Round Trip Time medio della rete è di 0,5 millisecondi, verranno spesi circa 50.000 millisecondi (50 secondi) per trasferire le richieste.

D'altra parte, se il valore di conteggio fosse 100.000, sarebbe necessaria una singola iterazione e solo 0,5 ms sarebbero spesi per trasferirla. Tuttavia, il motore sarebbe completamente bloccato per altre operazioni fino a quando il comando non termina di spazzare tutto il keyspace.

Oltre a KEYS, molti altri comandi sono potenzialmente dannosi se non utilizzati correttamente. Per visualizzare un elenco di tutti i comandi e la rispettiva complessità temporale, andare all'indirizzo <https://redis.io/commands>.

Esempi di potenziali problemi:

- Script Lua: Redis fornisce un interprete Lua incorporato, consentendo l'esecuzione di script sul lato server. Gli script Lua su Redis vengono eseguiti a livello di motore e sono atomici per definizione, ossia nessun altro comando o script avrà il permesso di eseguire mentre uno script è in esecuzione. Gli script Lua offrono la possibilità di eseguire più comandi, algoritmi decisionali, analisi dei dati e altri direttamente sul motore di Redis. Mentre l'atomicità degli script e la possibilità di scaricare l'applicazione sono allettanti, gli script devono essere utilizzati con cura e per piccole operazioni. Sì ElastiCache, il tempo di esecuzione degli script Lua

è limitato a 5 secondi. Gli script che non sono stati scritti nel keyspace verranno terminati automaticamente dopo un periodo di 5 secondi. Per evitare il danneggiamento dei dati e le incongruenze, il nodo eseguirà il failover se l'esecuzione dello script non è stata completata in 5 secondi e durante l'esecuzione non è stata eseguita alcuna scrittura. Le [Transazioni](#) sono l'alternativa per garantire la coerenza di più modifiche chiave correlate in Redis. Una transazione consente l'esecuzione di un blocco di comandi, osservando le chiavi esistenti per le modifiche. Se una delle chiavi osservate cambia prima del completamento della transazione, tutte le modifiche vengono scartate.

- Eliminazione di massa di elementi: il comando DEL accetta più parametri, che sono i nomi di chiave da eliminare. Le operazioni di eliminazione sono sincrone e richiedono un tempo significativo della CPU se l'elenco dei parametri è grande, o contiene un elenco grande, un set ordinato o un hash (strutture di dati che contengono più sottovoci). In altre parole, anche la cancellazione di una singola chiave può richiedere tempo significativo se ha molti elementi. L'alternativa a DEL è UNLINK, che è un comando asincrono disponibile da Redis 4. UNLINK deve essere preferito a DEL ogniqualvolta è possibile. A ElastiCache partire da Redis 6x, il `lazyfree-lazy-user-del` parametro fa sì che il DEL comando si comporti come quando è abilitato. UNLINK Per ulteriori informazioni, consultare la sezione [Modifica dei parametri di Redis 6.0](#).
- Comandi che agiscono su più chiavi: DEL è stato menzionato prima come comando che accetta più argomenti e il suo tempo di esecuzione sarà direttamente proporzionale a quello. Tuttavia, Redis fornisce molti altri comandi che funzionano in modo simile. Ad esempio, MSET e MGET consentono l'inserimento o il recupero di più chiavi di stringa contemporaneamente. Il loro utilizzo può essere utile per ridurre la latenza di rete intrinseca a più comandi SET o GET individuali. Tuttavia, un ampio elenco di parametri influenzerà l'utilizzo della CPU.

Sebbene l'utilizzo della CPU da solo non sia la causa di problemi di connettività, dedicare troppo tempo all'elaborazione di uno o pochi comandi su più chiavi può causare l'utilizzo fuori limite di altre richieste e aumentare l'utilizzo complessivo della CPU.

Il numero di chiavi e le loro dimensioni influenzeranno la complessità del comando e, di conseguenza, il tempo di completamento.

Altri esempi di comandi che possono agire su più chiavi: HMGET, HMSET, MSETNX, PFCOUNT, PFMERGE, SDIFF, SDIFFSTORE, SINTER, SINTERSTORE, SUNION, SUNIONSTORE, TOUCH, ZDIFF, ZDIFFSTORE, ZINTER o ZINTERSTORE.

- Comandi che agiscono su più tipi di dati: Redis fornisce anche comandi che agiscono su una o più chiavi, indipendentemente dal tipo di dati. ElastiCache for Redis fornisce la metrica

KeyBasedCmds per monitorare tali comandi. Questo parametro somma l'esecuzione dei seguenti comandi nel periodo selezionato:

- Complessità $O(N)$:
 - KEYS
- $O(1)$
 - EXISTS
 - OBJECT
 - PTTL
 - RANDOMKEY
 - TTL
 - TYPE
 - EXPIRE
 - EXPIREAT
 - MOVE
 - PERSIST
 - PEXPIRE
 - PEXPIREAT
 - UNLINK ($O(N)$) per recuperare memoria. Tuttavia il processo di recupero della memoria avviene in un thread separato e non blocca il motore
- Tempi di complessità differenti a seconda del tipo di dati:
 - DEL
 - DUMP
 - RENAME è considerato un comando con complessità $O(1)$, ma esegue DEL internamente. Il tempo di esecuzione varia a seconda delle dimensioni della chiave rinominata.
 - RENAMENX
 - RESTORE
 - SORT
- Big hash: Hash è un tipo di dati che consente una singola chiave con più sottoelementi chiave-valore. Ogni hash può archiviare 4.294.967.295 elementi e le operazioni su hash di grandi dimensioni possono diventare costose. Analogamente a KEYS, gli hash hanno il comando HKEYS con complessità temporale $O(N)$, dove N è il numero di elementi

nell'hash.HSCAN deve essere preferito a HKEYS per evitare comandi di lunga durata. HDEL, HGETALL, HMGET, HMSET e HVALS sono comandi che dovrebbero essere usati con cautela su hash di grandi dimensioni.

- Altre strutture Big Data: oltre agli hash, altre strutture di dati possono richiedere un uso intensivo della CPU. Insiemi, elenchi, insiemi ordinati e hyperloglog possono anche richiedere molto tempo per essere manipolati a seconda delle dimensioni e dei comandi utilizzati. Per ulteriori informazioni su tali comandi, consultare <https://redis.io/commands>.

Convalida della connettività di rete

Dopo aver esaminato le configurazioni di rete relative alla risoluzione del DNS, ai gruppi di sicurezza, alle ACL di rete e alle tabelle di routing, la connettività può essere convalidata con VPC Reachability Analyzer e gli strumenti di sistema.

Reachability Analyzer verificherà la connettività di rete e confermerà se tutti i requisiti e le autorizzazioni sono soddisfatti. Per i test seguenti è necessario l'ENI ID (Elastic Network Interface Identification) di uno dei ElastiCache nodi disponibili nel tuo VPC. È possibile trovarlo eseguendo le seguenti operazioni:

1. Andare su <https://console.aws.amazon.com/ec2/v2/home?#NIC:>
2. Filtra l'elenco delle interfacce in base al nome del ElastiCache cluster o all'indirizzo IP ottenuto dalle convalide DNS precedenti.
3. Annotare o comunque salvare l'ID ENI. Se vengono visualizzate più interfacce, rivedi la descrizione per confermare che appartengano al ElastiCache cluster corretto e scegline una.
4. Passare alla fase successiva.
5. Create un percorso di analisi su [https://console.aws.amazon.com/vpc/home? # ReachabilityAnalyzer](https://console.aws.amazon.com/vpc/home?#ReachabilityAnalyzer) e scegliete le seguenti opzioni:
 - Tipo di fonte: scegli l'istanza se il tuo ElastiCache client funziona su un'istanza Amazon EC2 o un'interfaccia di rete se utilizza un altro servizio, come AWS Fargate Amazon ECS con rete awsvpc AWS Lambda, ecc.) e il rispettivo ID di risorsa (istanza EC2 o ID ENI);
 - Tipo di destinazione: Scegliere Interfaccia di rete e selezionare il ENI di ElastiCachedall'elenco.
 - Porta di destinazione: specifica 6379 per Redis o 11211 ElastiCache per Memcached. ElastiCache Queste sono le porte definite con la configurazione di default e questo esempio presuppone che non vengano modificate.

- Protocollo: TCP

Creare il percorso di analisi e attendere alcuni minuti per il risultato. Se lo stato non è raggiungibile, aprire i dettagli dell'analisi ed esaminare Esplora analisi per i dettagli in cui le richieste sono state bloccate.

Se i test di raggiungibilità sono stati superati, procedere alla verifica a livello di sistema operativo.

Per convalidare la connettività TCP sulla porta di ElastiCache servizio: su Amazon Linux, Nping è disponibile nel pacchetto nmap e può testare la connettività TCP sulla ElastiCache porta, oltre a fornire il tempo di andata e ritorno della rete per stabilire la connessione. Usalo per convalidare la connettività di rete e la latenza corrente verso il cluster, come illustrato di seguito: ElastiCache

```
$ sudo nping --tcp -p 6379 example.xxxxxx.ng.0001.use1.cache.amazonaws.com
```

```
Starting Nping 0.6.40 ( http://nmap.org/nping ) at 2020-12-30 16:48 UTC
```

```
SENT (0.0495s) TCP ...
```

```
(Output suppressed )
```

```
Max rtt: 0.937ms | Min rtt: 0.318ms | Avg rtt: 0.449ms
```

```
Raw packets sent: 5 (200B) | Rcvd: 5 (220B) | Lost: 0 (0.00%)
```

```
Nping done: 1 IP address pinged in 4.08 seconds
```

Per impostazione predefinita, nping invia 5 sonde con un ritardo di 1 secondo tra di loro. È possibile utilizzare l'opzione “-c” per aumentare il numero di sonde e “--delay” per cambiare il tempo di invio di un nuovo test.

Se i test con nping falliscono e i test del VPC Reachability Analyzer passano, chiedere all'amministratore di sistema di esaminare le possibili regole firewall basate su host, regole di routing asimmetriche o qualsiasi altra possibile restrizione a livello di sistema operativo.

Sulla ElastiCache console, controlla se Encryption in-transit è abilitata nei dettagli del cluster ElastiCache . Se la crittografia in transito è abilitata, verificare se la sessione della TLS può essere stabilita con il seguente comando:

```
openssl s_client -connect example.xxxxxx.use1.cache.amazonaws.com:6379
```

Un output esteso è previsto se la connessione e la negoziazione relativa alla TLS hanno esito positivo. Controllare il codice restituito disponibile nell'ultima riga, il valore deve essere 0 (ok). Se

openssl restituisce qualcosa di diverso, controllare il motivo dell'errore su <https://www.openssl.org/docs/man1.0.2/man1/verify.html#DIAGNOSTICS>.

Se tutti i test dell'infrastruttura e del sistema operativo sono stati superati ma l'applicazione non riesce ancora a connettersi ElastiCache, verifica se le configurazioni dell'applicazione sono conformi alle impostazioni. ElastiCache Gli errori più comuni sono:

- L'applicazione non supporta la modalità ElastiCache cluster e la modalità cluster ElastiCache è abilitata;
- L'applicazione non supporta TLS/SSL e ElastiCache ha la crittografia in transito abilitata;
- L'applicazione supporta TLS/SSL ma non dispone dei flag di configurazione corretti o delle certification authority attendibili;

Limiti relativi alla rete

- Numero massimo di connessioni: ci sono limiti rigidi per le connessioni simultanee. Ogni ElastiCache nodo consente fino a 65.000 connessioni simultanee su tutti i client. Questo limite può essere monitorato tramite le `CurrConnections` metriche su CloudWatch Tuttavia, i client hanno anche i loro limiti per le connessioni in uscita. Su Linux, controllare l'intervallo di porte effimere consentito con il comando:

```
# sysctl net.ipv4.ip_local_port_range
net.ipv4.ip_local_port_range = 32768 60999
```

Nell'esempio precedente, saranno consentite 28231 connessioni dalla stessa origine, allo stesso IP (ElastiCache nodo) e porta di destinazione. Il comando seguente mostra quante connessioni esistono per un ElastiCache nodo specifico (IP 1.2.3.4):

```
ss --numeric --tcp state connected "dst 1.2.3.4 and dport == 6379" | grep -vE
'^State' | wc -l
```

Se il numero è troppo alto, il sistema potrebbe sovraccaricarsi tentando di elaborare le richieste di connessione. Si consiglia di prendere in considerazione l'implementazione di tecniche come il pool di connessioni o connessioni persistenti per gestire meglio le connessioni. Ogniqualvolta è possibile, configurare il pool di connessioni per limitare il numero massimo di connessioni ad alcune centinaia. Inoltre, la logica di back-off per gestire i timeout o altre eccezioni relative alla connessione sono consigliabili per evitare interruzioni di connessione in caso di problemi.

- Limiti del traffico di rete: controlla le seguenti [CloudWatch metriche per Redis per](#) identificare i possibili limiti di rete raggiunti sul nodo: ElastiCache
 - `NetworkBandwidthInAllowanceExceeded` / `NetworkBandwidthOutAllowanceExceeded`: pacchetti di rete sagomati perché la velocità effettiva ha superato il limite di larghezza di banda aggregata.

È importante notare che ogni byte scritto nel nodo primario verrà replicato in N repliche, dove N è il numero di repliche. I cluster con tipi di nodi piccoli, repliche multiple e richieste di scrittura ad alta intensità potrebbero non essere in grado di gestire il backlog di replica. In questi casi, è best practice dimensionare verso l'alto (cambiare il tipo di nodo), aumentare orizzontalmente (aggiungere partizioni in cluster in modalità cluster abilitata), ridurre il numero di repliche o ridurre al minimo il numero di scritture.

- `NetworkConntrackAllowanceExceeded`: pacchetti sagomati perché è stato superato il numero massimo di connessioni monitorate in tutti i gruppi di sicurezza assegnati al nodo. Le nuove connessioni probabilmente non avranno esito positivo durante questo periodo.
- `NetworkPacketsPerSecondAllowanceExceeded`: Numero massimo di pacchetti al secondo superato. I carichi di lavoro basati su un elevato tasso di richieste molto piccole possono raggiungere questo limite prima della larghezza di banda massima.

I parametri sopra riportati sono il modo ideale per confermare che i nodi raggiungono i loro limiti di rete. Tuttavia, i limiti sono identificabili anche dai plateau sui parametri di rete.

Se i plateau vengono osservati per periodi prolungati, saranno probabilmente seguiti da ritardo di replica, aumento dei byte utilizzati per la cache, calo della memoria liberabile, elevato swap e utilizzo della CPU. Le istanze Amazon EC2 hanno anche limiti di rete che possono essere monitorati attraverso [Parametri di driver ENA](#). Le istanze Linux con supporto di rete avanzato e driver ENA 2.2.10 o versioni successive possono esaminare i contatori del limite con il comando:

```
# ethtool -S eth0 | grep "allowance_exceeded"
```

Utilizzo CPU

La metrica di utilizzo della CPU è il punto di partenza dell'indagine e i seguenti elementi possono aiutare a restringere i possibili problemi secondari: ElastiCache

- **Redis SlowLogs**: la configurazione ElastiCache predefinita mantiene gli ultimi 128 comandi che hanno richiesto più di 10 millisecondi per essere completati. La cronologia dei comandi lenti viene

mantenuta durante il tempo di esecuzione del motore e andrà persa in caso di guasto o riavvio. Se l'elenco raggiunge 128 voci, i vecchi eventi verranno rimossi per fare spazio a quelli nuovi. La dimensione dell'elenco degli eventi lenti e il tempo di esecuzione considerato lento può essere modificato tramite i parametri `slowlog-max-len` e `slowlog-log-slower-than` in un [gruppo di parametri personalizzati](#). L'elenco dei slow logs può essere recuperato eseguendo `SLOWLOG GET` 128 sul motore, in cui 128 sono gli ultimi 128 comandi lenti segnalati. Ogni voce ha i campi riportati di seguito:

```
1) 1) (integer) 1 -----> Sequential ID
   2) (integer) 1609010767 --> Timestamp (Unix epoch time)of the Event
   3) (integer) 4823378 -----> Time in microseconds to complete the command.
   4) 1) "keys" -----> Command
      2) "*" -----> Arguments
   5) "1.2.3.4:57004"-> Source
```

L'evento di cui sopra è avvenuto il 26 dicembre, alle 19:26:07 UTC, ha impiegato 4,8 secondi (4,823 ms) per essere completato ed è stato causato dal comando KEYS richiesto dal client 1.2.3.4.

Su Linux, la marcatura oraria può essere convertita con la data del comando:

```
$ date --date='@1609010767'
Sat Dec 26 19:26:07 UTC 2020
```

Con Python:

```
>>> from datetime import datetime
>>> datetime.fromtimestamp(1609010767)
datetime.datetime(2020, 12, 26, 19, 26, 7)
```

Oppure su Windows con: PowerShell

```
PS D:\Users\user> [datetimeoffset]::FromUnixTimeSeconds('1609010767')
DateTime           : 12/26/2020 7:26:07 PM
UtcDateTime        : 12/26/2020 7:26:07 PM
LocalDateTime      : 12/26/2020 2:26:07 PM
Date               : 12/26/2020 12:00:00 AM
Day                : 26
DayOfWeek          : Saturday
```

```

DayOfYear      : 361
Hour           : 19
Millisecond    : 0
Minute        : 26
Month
                : 12
Offset        : 00:00:00Ticks           : 637446075670000000
UtcTicks
                : 637446075670000000
TimeOfDay     : 19:26:07
Year          : 2020

```

Molti comandi lenti in un breve periodo di tempo (stesso minuto o meno) sono motivo di preoccupazione. Esaminare la natura dei comandi e come possono essere ottimizzati (vedere esempi precedenti). Se i comandi con complessità temporale $O(1)$ vengono segnalati frequentemente, controllare gli altri fattori relativamente a un utilizzo elevato della CPU come menzionato in precedenza.

- **Metriche di latenza:** ElastiCache for Redis fornisce CloudWatch metriche per monitorare la latenza media per diverse classi di comandi. Il datapoint viene calcolato dividendo il numero totale di esecuzioni di comandi nella categoria per il tempo totale di esecuzione nel periodo. È importante comprendere che i risultati dei parametri di latenza sono un aggregato di più comandi. Un singolo comando può causare risultati imprevisti, come i timeout, senza un impatto significativo sui parametri. In questi casi, gli eventi di slow log sarebbero una fonte di informazioni più accurata. L'elenco seguente contiene i parametri di latenza disponibili e i rispettivi comandi che le riguardano.
 - **EvalBasedCmdsLatency:** relativo ai comandi Lua Script,,; `eval evalsha`
 - **GeoSpatialBasedCmdsLatency:** `geodist, geohash, geopos, georadius, georadiusbymember, geoadd;`
 - **GetTypeCmdsLatency:** comandi di lettura, indipendentemente dal tipo di dati;
 - **HashBasedCmdsLatency:** `hexists, hget, hgetall, hkeys, hlen, hmget, hvals, hstrlen, hdel, hincrby, hincrbyfloat, hmset, hset, hsetnx;`
 - **HyperLogLogBasedCmdsLatency:** `pfselftest, pfcount, pfdebug, pfadd, pfmerge;`
 - **KeyBasedCmdsLatency:** Comandi che possono agire su diversi tipi di dati: `dump existskeys,object,pttl,,randomkey,ttl,type,del,expire,expireat,move,persist,expire`
 - **ListBasedCmdsLatency:** `lindex, len, range, blpop, brpop, brpoplpush, linsert, lpop, push, lpushx, lrem, lset, ltrim, rpop, rpoplpush, rpush, rpushx;`

- **PubSubBasedCmdsLatency:** psubscribe, publish, pubsub, annulla sottoscrizione, sottoscrizione, annullamento dell'iscrizione;
- **SetBasedCmdsLatency:** scard, sdiff, sinter, sismember, smembers, srandmember, sunion, sadd, sdiffstore, sinterstore, smove, spop, srem, sunionstore;
- **SetTypeCmdsLatency:** scrivere comandi, indipendentemente dal tipo di dati;
- **SortedSetBasedCmdsLatency:** zcard, zcount, zrange, zrangebyscore, zrank, zrevrange, zrevrangebyscore, zrevrank, zscore, zrangebylex, zrevrangebylex, zlexcount, zadd, zincrby, zinterstore, zrem, zremrangebyrank, zremrangebyscore, zunionstore, zremrangebylex, zpopmax, zpopmin, bzpopmin, bzpopmax;
- **StringBasedCmdsLatency:** bitcount, get, getbit, getrange, mget, strlen, substr, bitpos, append, bitop, bitfield, decr, decrby, getset, incr, incrby, incrbyfloat, mset, msetnx, psetex, set, setbit, setex, setnx, setrange;
- **StreamBasedCmdsLatency:** xrange, xrevrange, xlen, xread, xpending, xinfo, xadd, xgroup, readgroup, xack, xclaim, xdel, xtrim, xsetid;
- **Comandi relativi al tempo di esecuzione di Redis:**
 - **info commandstats:** fornisce un elenco di comandi eseguiti dall'avvio del motore Redis, il numero di esecuzioni cumulative, il tempo totale di esecuzione e il tempo medio di esecuzione per comando;
 - **client list:** fornisce un elenco di client attualmente connessi e informazioni rilevanti come l'utilizzo dei buffer, l'ultimo comando eseguito, ecc.;
- **Backup e replica:** ElastiCache per le versioni Redis precedenti alla 2.8.22, utilizza un processo biforcuto per creare backup ed elaborare sincronizzazioni complete con le repliche. Questo metodo può incorrere in un sovraccarico di memoria significativo per casi d'uso intensivi di scrittura.

A partire da ElastiCache Redis 2.8.22, è stato introdotto un metodo di backup e replica senza forkless. AWS Il nuovo metodo potrebbe ritardare le scritture al fine di evitare errori. Entrambi i metodi possono causare periodi di maggiore utilizzo della CPU, portare a tempi di risposta più elevati e di conseguenza portare a timeout client durante la loro esecuzione. Controllare sempre se gli errori del client si verificano durante la finestra di backup o il parametro SaveInProgress era 1 nel periodo. Si consiglia di pianificare la finestra di backup per periodi di basso utilizzo per ridurre al minimo la possibilità di problemi con i client o errori di backup.

Connessioni terminate dal lato server

L'impostazione predefinita ElastiCache per la configurazione Redis mantiene le connessioni client stabilite a tempo indeterminato. Tuttavia, in alcuni casi la terminazione della connessione potrebbe essere auspicabile. Esempio:

- I bug nell'applicazione client possono causare connessioni da dimenticare e mantenere stabilite con uno stato di inattività. Questo è chiamato “perdita di connessione” e la conseguenza è un costante aumento del numero di connessioni stabilite osservate sul parametro `CurrentConnections`. Questo comportamento può comportare una saturazione del client o del lato. ElastiCache Quando una correzione immediata non è possibile dal lato client, alcuni amministratori impostano un valore di «timeout» nel proprio ElastiCache gruppo di parametri. Il timeout è il tempo in secondi consentito per la persistenza delle connessioni inattive. Se il client non invia alcuna richiesta nel periodo, il motore Redis interromperà la connessione non appena la connessione raggiunge il valore di timeout. Piccoli valori di timeout possono causare disconnessioni non necessarie e i client dovranno gestirle correttamente e riconnettersi, causando ritardi.
- La memoria utilizzata per memorizzare le chiavi è condivisa con i buffer client. I client lenti con richieste o risposte di grandi dimensioni possono richiedere una quantità significativa di memoria per gestire i buffer. L'impostazione predefinita ElastiCache per le configurazioni Redis non limita la dimensione dei normali buffer di output del client. Se il limite `maxmemory` è raggiunto, il motore tenterà di espellere gli elementi per soddisfare l'utilizzo del buffer. In condizioni di memoria estremamente ridotta, ElastiCache For Redis potrebbe scegliere di disconnettere i client che consumano buffer di output client di grandi dimensioni per liberare memoria e mantenere lo stato del cluster.

È possibile limitare la dimensione dei buffer client con configurazioni personalizzate e i client che raggiungono il limite verranno disconnessi. Tuttavia, i client dovrebbero essere in grado di gestire disconnessioni impreviste. I parametri per gestire le dimensioni dei buffer per i client regolari sono i seguenti:

- `client-query-buffer-limit`: dimensione massima di una singola richiesta di input;
- `client-output-buffer-limit-normal-soft-limit`: Limite flessibile per le connessioni client. La connessione verrà interrotta se rimane al di sopra del limite libero per più del tempo in secondi definito su `client-output-buffer-limit-normal-soft-seconds` o se raggiunge il limite rigido;
- `client-output-buffer-limit-normal-soft-seconds`: Tempo consentito per le connessioni che superano il `client-output-buffer-limit -; normal-soft-limit`

- `client-output-buffer-limit-normal-hard-limit`: Una connessione che raggiunge questo limite verrà immediatamente interrotta.

Oltre ai buffer client normali, le seguenti opzioni controllano il buffer per i nodi di replica e i client Pub/Sub (Publish/Subscribe):

- `client-output-buffer-limit-replica-hard-limit`;
- `client-output-buffer-limit-replica-soft-seconds`;
- `client-output-buffer-limit-replica-hard-limit`;
- `client-output-buffer-limit-pubsub-soft-limit`;
- `client-output-buffer-limit-pubsub-soft-seconds`;
- `client-output-buffer-limit-pubsub-hard-limit`;

Risoluzione dei problemi lato client per istanze Amazon EC2

Il carico e la reattività sul lato client possono influire anche sulle richieste di ElastiCache. I limiti dell'istanza EC2 e del sistema operativo devono essere attentamente esaminati durante la risoluzione dei problemi di connettività intermittente o timeout. Alcuni punti chiave da osservare:

- CPU:
 - Utilizzo della CPU dell'istanza EC2: assicurarsi che la CPU non sia stata saturata o vicina al 100 per cento. L'analisi storica può essere effettuata tramite CloudWatch, tuttavia, tieni presente che la granularità dei punti dati è di 1 minuto (con il monitoraggio dettagliato abilitato) o di 5 minuti;
 - Se si utilizza [istanze EC2 burstable](#), assicurarsi che il loro saldo di credito della CPU non sia stato esaurito. Queste informazioni sono disponibili sulla metrica `CPUCreditBalance` CloudWatch
 - Brevi periodi di utilizzo elevato della CPU possono causare timeout senza che ciò comporti un utilizzo al 100%. CloudWatch Tali casi richiedono un monitoraggio in tempo reale con strumenti del sistema operativo come `top`, `ps` e `mpstat`.
- Rete
 - Verificare se la velocità effettiva di rete è inferiore ai valori accettabili in base alle capacità dell'istanza. Per ulteriori informazioni consultare [Tipi di istanza Amazon EC2](#)
 - Sulle istanze con ena Driver di rete avanzato, controllare le [Statistiche ENA](#) per i timeout o i limiti superati. Le seguenti statistiche sono utili per verificare la saturazione dei limiti di rete:

- `bw_in_allowance_exceeded / bw_out_allowance_exceeded`: numero di pacchetti sagomati a causa di una eccessiva velocità effettiva in entrata o in uscita;
- `contrack_allowance_exceeded`: numero di pacchetti eliminati a causa dei [Limiti di monitoraggio delle connessioni](#) dei gruppi di sicurezza. Le nuove connessioni avranno esito negativo quando questo limite è saturo;
- `linklocal_allowance_exceeded`: numero di pacchetti eliminati a causa di richieste eccessive per l'istanza di meta-dati, NTP tramite DNS VPC. Il limite è di 1024 pacchetti al secondo per tutti i servizi;
- `pps_allowance_exceeded`: numero di pacchetti eliminati a causa di un rapporto eccessivo di pacchetti al secondo. Il limite PPS può essere raggiunto quando il traffico di rete è costituito da migliaia o milioni di richieste molto piccole al secondo. ElastiCache il traffico può essere ottimizzato per utilizzare meglio i pacchetti di rete tramite pipeline o comandi che eseguono più operazioni contemporaneamente, come invece di. MGET GET

Dissezione del tempo necessario per completare una singola richiesta

- In rete: `Tcpdump` and `Wireshark` (`tshark` sulla riga di comando) sono strumenti utili per capire quanto tempo ha impiegato la richiesta per percorrere la rete, avviare il ElastiCache motore e ottenere un ritorno. L'esempio seguente evidenzia una singola richiesta creata con il seguente comando:

```
$ echo ping | nc example.xxxxxx.ng.0001.use1.cache.amazonaws.com 6379
+PONG
```

In parallelo al comando di cui sopra, `tcpdump` era in esecuzione ed è stato restituito:

```
$ sudo tcpdump -i any -nn port 6379 -tt
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on any, link-type LINUX_SLL (Linux cooked), capture size 262144 bytes
1609428918.917869 IP 172.31.11.142.40966
    > 172.31.11.247.6379: Flags [S], seq 177032944, win 26883, options [mss
    8961,sackOK,TS val 27819440 ecr 0,nop,wscale 7], length 0
1609428918.918071 IP 172.31.11.247.6379 > 172.31.11.142.40966: Flags [S.], seq
    53962565, ack 177032945, win
    28960, options [mss 1460,sackOK,TS val 3788576332 ecr 27819440,nop,wscale 7],
    length 0
```

```

1609428918.918091 IP 172.31.11.142.40966 > 172.31.11.247.6379: Flags [.], ack 1, win
 211, options [nop,nop,TS val 27819440 ecr 3788576332], length 0
1609428918.918122
  IP 172.31.11.142.40966 > 172.31.11.247.6379: Flags [P.], seq 1:6, ack 1, win 211,
  options [nop,nop,TS val 27819440 ecr 3788576332], length 5: RESP "ping"
1609428918.918132 IP 172.31.11.142.40966 > 172.31.11.247.6379: Flags [F.], seq 6, ack
 1, win 211, options [nop,nop,TS val 27819440 ecr 3788576332], length 0
1609428918.918240 IP 172.31.11.247.6379 > 172.31.11.142.40966: Flags [.], ack 6, win
 227, options [nop,nop,TS val 3788576332 ecr 27819440], length 0
1609428918.918295
  IP 172.31.11.247.6379 > 172.31.11.142.40966: Flags [P.], seq 1:8, ack 7, win 227,
  options [nop,nop,TS val 3788576332 ecr 27819440], length 7: RESP "PONG"
1609428918.918300 IP 172.31.11.142.40966 > 172.31.11.247.6379: Flags [.], ack 8, win
 211, options [nop,nop,TS val 27819441 ecr 3788576332], length 0
1609428918.918302 IP 172.31.11.247.6379 > 172.31.11.142.40966: Flags [F.], seq 8, ack
 7, win 227, options [nop,nop,TS val 3788576332 ecr 27819440], length 0
1609428918.918307
  IP 172.31.11.142.40966 > 172.31.11.247.6379: Flags [.], ack 9, win 211, options
  [nop,nop,TS val 27819441 ecr 3788576332], length 0
^C
10 packets captured
10 packets received by filter
0 packets dropped by kernel

```

Dall'output di cui sopra possiamo confermare che l'handshake a tre vie TCP è stato completato in 222 microsecondi (918091 - 917869) e che il comando ping è stato inviato e restituito in 173 microsecondi (918295 - 918122).

Ci sono voluti 438 microsecondi (918307 - 917869) dalla richiesta alla chiusura della connessione. Tali risultati confermerebbero che i tempi di risposta della rete e del motore sono buoni e l'indagine può concentrarsi su altri componenti.

- Nel sistema operativo: Strace può aiutare a identificare le lacune temporali a livello di sistema operativo. L'analisi delle applicazioni effettive sarebbe molto più ampia e sono consigliati profiler di applicazioni specializzati o debugger. L'esempio seguente mostra solo se i componenti del sistema operativo di base funzionano come previsto, in caso contrario potrebbero essere necessarie ulteriori indagini. Utilizzando lo stesso comando PING di Redis con `strace` otteniamo:

```

$ echo ping | strace -f -tttt -r -e trace=execve,socket,open,recvfrom,sendto
nc example.xxxxxx.ng.0001.use1.cache.amazonaws.com (http://
example.xxxxxx.ng.0001.use1.cache.amazonaws.com/)
6379

```


Sicurezza in Amazon ElastiCache

Per AWS, la sicurezza del cloud ha la massima priorità. In quanto cliente AWS, è possibile trarre vantaggio da un'architettura di data center e di rete progettata per soddisfare i requisiti delle organizzazioni più esigenti a livello di sicurezza.

La sicurezza è una responsabilità condivisa tra te e AWS. Il [modello di responsabilità condivisa](#) descrive questo come sicurezza del cloud e sicurezza nel cloud:

- La sicurezza del cloud: AWS è responsabile della protezione dell'infrastruttura che gestisce i servizi AWS nel cloud AWS. AWS fornisce inoltre servizi che puoi utilizzare in sicurezza. I revisori di terze parti testano e verificano regolarmente l'efficacia della sicurezza come parte dei [programmi di conformità AWS](#). Per ulteriori informazioni sui programmi di conformità che si applicano ad Amazon ElastiCache, consulta [AWS Servizi coperti dal programma di conformità](#).
- Sicurezza nel cloud: la tua responsabilità è determinata dal servizio AWS che utilizzi. Sei anche responsabile di altri fattori, tra cui la riservatezza dei dati, i requisiti della tua azienda e le leggi e normative vigenti.

Questa documentazione consente di comprendere come applicare il modello di responsabilità condivisa quando si usa Amazon ElastiCache. Gli argomenti seguenti descrivono come configurare Amazon ElastiCache per soddisfare gli obiettivi di sicurezza e conformità. Vengono inoltre fornite informazioni su come utilizzare altri servizi AWS che consentono di monitorare e proteggere le risorse Amazon ElastiCache.

Argomenti

- [Protezione dei dati in Amazon ElastiCache](#)
- [Riservatezza del traffico Internet](#)
- [Identity and Access Management per Amazon ElastiCache](#)
- [Convalida della conformità per Amazon ElastiCache](#)
- [Resilienza in Amazon ElastiCache](#)
- [Sicurezza dell'infrastruttura in AWS ElastiCache](#)
- [Aggiornamenti del servizio in ElastiCache](#)

Protezione dei dati in Amazon ElastiCache

Il [modello di responsabilità condivisa](#) AWS si applica alla protezione dei dati in AWS ElastiCache. Come descritto in questo modello, AWS è responsabile della protezione dell'infrastruttura globale che esegue l'intero cloud AWS. L'utente è responsabile di mantenere il controllo sui contenuti ospitati su questa infrastruttura. Questo contenuto include la configurazione della protezione e le attività di gestione per i servizi AWS utilizzati. Per ulteriori informazioni sulla privacy dei dati, consulta [Domande frequenti sulla privacy dei dati](#).

Per garantire la protezione dei dati, si consiglia di proteggere le credenziali dell'account AWS e di configurare singoli account con AWS Identity and Access Management (IAM). In questo modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere il proprio lavoro. Ti suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- Utilizza TLS per comunicare con risorse AWS.
- Configura la creazione di log delle attività di API e utenti con AWS CloudTrail.
- Utilizza le soluzioni di crittografia AWS, insieme a tutti i controlli di sicurezza predefiniti all'interno dei servizi AWS.
- Utilizza i servizi di sicurezza gestiti avanzati, ad esempio Amazon Macie, che aiutano a individuare e proteggere i dati personali archiviati in Amazon S3.

Ti consigliamo di non inserire mai informazioni identificative sensibili, ad esempio i numeri di account dei clienti, in campi a formato libero, ad esempio un campo Name (Nome). Questo suggerimento si applica all'utilizzo di ElastiCache o altri servizi AWS con la console, l'API, la AWS CLI o gli SDK AWS. Gli eventuali dati immessi in ElastiCache o altri servizi possono essere prelevati per l'inserimento nei log di diagnostica. Quando fornisci un URL a un server esterno, non includere informazioni sulle credenziali nell'URL per convalidare la tua richiesta a tale server.

Argomenti

- [Sicurezza dei dati in Amazon ElastiCache](#)

Sicurezza dei dati in Amazon ElastiCache

Per mantenere la sicurezza dei dati, Amazon ElastiCache e Amazon EC2 forniscono meccanismi di protezione da accessi non autorizzati dei dati sul server.

Amazon ElastiCache per Memcached fornisce anche funzionalità di crittografia per i dati su cache che eseguono Memcached 1.6.12 o versioni successive.

- La crittografia dei dati in transito viene eseguita quando i dati si spostano da una posizione a un'altra, ad esempio tra i nodi del cluster o tra la cache e l'applicazione.
- La crittografia dei dati inattivi esegue la crittografia su disco dei dati durante operazioni di sincronizzazione e backup.

Argomenti

- [Crittografia in transito ElastiCache \(TLS\)](#)
- [Crittografia dei dati a riposo in ElastiCache](#)

Crittografia in transito ElastiCache (TLS)

Per mantenere la sicurezza dei dati, Amazon ElastiCache e Amazon EC2 forniscono meccanismi di protezione da accessi non autorizzati dei dati sul server. Grazie alla caratteristica di crittografia dei dati in transito fornita da ElastiCache, disponi di uno strumento che puoi utilizzare per proteggere i dati quando si spostano da una posizione a un'altra.

Tutte le cache serverless dispongono della crittografia dei dati in transito abilitata. Per i cluster progettati autonomamente puoi abilitare la crittografia dei dati in transito su un cluster di cache impostando il parametro `TransitEncryptionEnabled` su `true` (CLI: `--transit-encryption-enabled`) durante la creazione del cluster di cache utilizzando l'operazione `CreateCacheCluster` (CLI: `create-cache-cluster`).

Argomenti

- [Panoramica della crittografia dei dati in transito](#)
- [Condizioni di crittografia dei dati in transito](#)
- [Best practice per la crittografia in transito](#)
- [Abilitazione della crittografia dei dati in transito](#)
- [Connessione a nodi con crittografia in transito abilitata tramite Openssl](#)
- [Creazione di un client TLS Memcached utilizzando Java](#)
- [Creazione di un client TLS Memcached utilizzando PHP](#)

Panoramica della crittografia dei dati in transito

La crittografia dei dati in transito di Amazon ElastiCache è una funzionalità che ti consente di aumentare la sicurezza dei tuoi dati nei punti più vulnerabili, quando sono in transito da una posizione all'altra. Poiché la crittografia e la decrittografia dei dati richiede l'elaborazione a livello di endpoint, l'abilitazione della crittografia dei dati in transito può in parte influire sulle prestazioni. È opportuno creare un riferimento per i dati con o senza crittografia dei dati in transito per determinare l'impatto sulle prestazioni per i propri casi d'uso.

La crittografia dei dati in transito ElastiCache implementa le seguenti caratteristiche:

- Connessioni client crittografate: le connessioni client ai nodi di cache sono crittografate con TLS.
- Connessioni server crittografate: i dati che si spostano tra i nodi di un cluster sono crittografati.
- Autenticazione del server: i client possono autenticare che si stanno connettendo al server giusto.

Condizioni di crittografia dei dati in transito

Tieni presente i seguenti vincoli sulla crittografia dei dati in transito di Amazon ElastiCache durante la pianificazione dell'implementazione di un cluster progettato autonomamente:

- La crittografia in transito è supportata su cluster che eseguono Memcached versione 1.6.12 e successive.
- La crittografia dei dati in transito supporta Transport Layer Security (TLS) versioni 1.2 e 1.3.
- La crittografia in transito è supportata solo per cluster in esecuzione in un VPC Amazon.
- La crittografia in transito è supportata solo per cluster che eseguono i tipi di nodi seguenti.
 - R6g, R5, R4
 - M6g, M5, M4
 - T4g, T3

Per ulteriori informazioni, consulta [Tipi di nodi supportati](#).

- La crittografia dei dati in transito è abilitata impostando in maniera esplicita il parametro `TransitEncryptionEnabled` su `true`.
- Puoi abilitare la crittografia in transito su un cluster solo durante la creazione del cluster. Non puoi attivare o disattivare la crittografia in transito modificando un cluster.
- Assicurati che il client di cache supporti la connettività TLS e che sia abilitata nella configurazione del client.

Best practice per la crittografia in transito

- A causa dell'elaborazione richiesta per crittografare e decrittografare i dati a livello degli endpoint, l'implementazione della crittografia dei dati in transito può ridurre le prestazioni. Raffronta la crittografia dei dati in transito con l'assenza di crittografia sui dati per determinare il suo impatto sulle prestazioni per l'implementazione.
- Siccome la creazione di nuove connessioni può richiedere molte risorse, puoi ridurre l'impatto della crittografia in transito sulle prestazioni mantenendo le connessioni SSL.

Abilitazione della crittografia dei dati in transito

Per abilitare la crittografia in transito creando un cluster Memcached tramite AWS Management Console, effettua le selezioni seguenti:

- Scegli Memcached come motore.
- Scegli la versione del motore 1.6.12 o successiva.
- In Encryption in transit (Crittografia in transito), scegli Enable (Abilita).

Per il processo dettagliato, consulta [Creazione di un cluster Memcached \(console\)](#).

Connessione a nodi con crittografia in transito abilitata tramite Openssl

Per l'accesso ai dati da nodi ElastiCache for Memcached con crittografia in transito abilitata, devi utilizzare client che funzionano con Secure Socket Layer (SSL). Su Amazon linux e Amazon Linux 2, puoi utilizzare anche Openssl s_client.

Per utilizzare Openssl s_client per la connessione a un cluster Memcached con crittografia in transito abilitata su Amazon Linux 2 o Amazon Linux:

```
/usr/bin/openssl s_client -connect memcached-node-endpoint:memcached-port
```

Creazione di un client TLS Memcached utilizzando Java

Per creare un client in modalità TLS, effettua le operazioni seguenti per inizializzare il client con l'SSLContext appropriato:

```
import java.security.KeyStore;  
import javax.net.ssl.SSLContext;  
import javax.net.ssl.TrustManagerFactory;
```

```
import net.spy.memcached.AddrUtil;
import net.spy.memcached.ConnectionFactoryBuilder;
import net.spy.memcached.MemcachedClient;
public class TLSDemo {
    public static void main(String[] args) throws Exception {
        ConnectionFactoryBuilder connectionFactoryBuilder = new
ConnectionFactoryBuilder();
        // Build SSLContext
        TrustManagerFactory tmf =
TrustManagerFactory.getInstance(TrustManagerFactory.getDefaultAlgorithm());
        tmf.init((KeyStore) null);
        SSLContext sslContext = SSLContext.getInstance("TLS");
        sslContext.init(null, tmf.getTrustManagers(), null);
        // Create the client in TLS mode
        connectionFactoryBuilder.setSSLContext(sslContext);
        MemcachedClient client = new MemcachedClient(connectionFactoryBuilder.build(),
AddrUtil.getAddresses("mycluster.fnjyzo.cfg.use1.cache.amazonaws.com:11211"));

        // Store a data item for an hour.
        client.set("theKey", 3600, "This is the data value");
    }
}
```

Creazione di un client TLS Memcached utilizzando PHP

Per creare un client in modalità TLS, effettua le operazioni seguenti per inizializzare il client con l'SSLContext appropriato:

```
<?php

/**
 * Sample PHP code to show how to create a TLS Memcached client. In this example we
 * will use the Amazon ElastiCache Auto Discovery feature, but TLS can also be
 * used with a Static mode client.
 * See Using the ElastiCache Cluster Client for PHP (https://docs.aws.amazon.com/AmazonElastiCache/latest/mem-ug/AutoDiscovery.Using.ModifyApp.PHP.html) for more
 * information
 * about Auto Discovery and persistent-id.
 */

/* Configuration endpoint to use to initialize memcached client.
 * this is only an example */
$server_endpoint = "mycluster.fnjyzo.cfg.use1.cache.amazonaws.com";
```

```
/* Port for connecting to the cluster.
 * This is only an example */
$server_port = 11211;

/* Initialize a persistent Memcached client and configure it with the Dynamic client
mode */
$tls_client = new Memcached('persistent-id');
$tls_client->setOption(Memcached::OPT_CLIENT_MODE, Memcached::DYNAMIC_CLIENT_MODE);

/* Add the memcached's cluster server/s */
$tls_client->addServer($server_endpoint, $server_port);

/* Configure the client to use TLS */
if(!$tls_client->setOption(Memcached::OPT_USE_TLS, 1)) {
    echo $tls_client->getLastErrorMessage(), "\n";
    exit(1);
}

/* Set your TLS context configurations values.
 * See MemcachedTLSContextConfig in memcached-api.php for all configurations */
$tls_config = new MemcachedTLSContextConfig();
$tls_config->hostname = '*.mycluster.fnjyzo.use1.cache.amazonaws.com';
$tls_config->skip_cert_verify = false;
$tls_config->skip_hostname_verify = false;

/* Use the created TLS context configuration object to create OpenSSL's SSL_CTX and set
it to your client.
 * Note: These TLS context configurations will be applied to all the servers connected
to this client. */
$tls_client->createAndSetTLSContext((array)$tls_config);

/* test the TLS connection with set-get scenario: */

/* store the data for 60 seconds in the cluster.
 * The client will decide which cache host will store this item.
 */
if($tls_client->set('key', 'value', 60)) {
    print "Successfully stored key\n";
} else {
    echo "Failed to set key: ", $tls_client->getLastErrorMessage(), "\n";
    exit(1);
}
```



```
/* retrieve the key */
if ($tls_client->get('key') === 'value') {
    print "Successfully retrieved key\n";
} else {
    echo "Failed to get key: ", $tls_client->getLastErrorMessage(), "\n";
    exit(1);
}
```

Per ulteriori informazioni sull'utilizzo del client PHP, consulta [Installazione del client del cluster ElastiCache per PHP](#).

Crittografia dei dati a riposo in ElastiCache

Per mantenere la sicurezza dei dati, Amazon ElastiCache e Amazon S3 forniscono metodi diversi per limitare l'accesso ai dati nella cache. Per ulteriori informazioni, consulta [Amazon VPC e sicurezza di ElastiCache](#) e [Identity and Access Management per Amazon ElastiCache](#).

- Il disco durante le operazioni di sincronizzazione e swap

ElastiCache offre la crittografia dei dati a riposo predefinita (servizio gestito), nonché la possibilità di utilizzare le chiavi simmetriche AWS KMS gestite dal cliente nel [Sistema AWS di gestione delle chiavi](#). Quando viene eseguito il backup della cache, nelle opzioni di crittografia scegli se utilizzare la chiave di crittografia predefinita o una chiave gestita dal cliente. Per ulteriori informazioni, consulta [Abilitazione della crittografia dei dati inattivi](#).

Note

La crittografia predefinita (gestita dal servizio) è l'unica opzione disponibile nelle regioni GovCloud (Stati Uniti).

La crittografia dei dati a riposo può essere abilitata su una cache solo quando viene creata. Poiché la crittografia e la decrittografia dei dati richiede l'elaborazione, l'abilitazione della crittografia dei dati inattivi può in parte influire sulle prestazioni durante queste operazioni. È opportuno creare un riferimento per i dati con o senza crittografia dei dati inattivi per determinare l'impatto sulle prestazioni per i propri casi d'uso.

Argomenti

- [Condizioni di crittografia dei dati inattivi](#)
- [Utilizzo di chiavi gestite dal cliente da KMS AWS](#)
- [Abilitazione della crittografia dei dati inattivi](#)
- [Vedi anche](#)

Condizioni di crittografia dei dati inattivi

Tieni presente i seguenti vincoli sulla crittografia dei dati inattivi di ElastiCache durante la pianificazione della crittografia dei dati inattivi di ElastiCache:

- La crittografia dei dati a riposo è supportata solo nelle cache serverless.
- Inoltre, l'opzione di utilizzare le chiavi gestite dal cliente per la crittografia dei dati inattivi non è disponibile nelle regioni AWS GovCloud (us-gov-east-1 e us-gov-west-1).

Utilizzo di chiavi gestite dal cliente da KMS AWS

ElastiCache supporta le chiavi simmetriche AWS KMS (chiavi KMS) gestite dal cliente per la crittografia di dati a riposo. Le chiavi KMS gestite dal cliente sono chiavi crittografiche che è possibile creare, possedere e gestire all'interno del proprio account AWS. Per ulteriori informazioni, consulta [AWSChiavi KMS](#) nella AWS Key Management Service Developer Guide (Guida Sviluppatore del Servizio di Gestione Chiave). Le chiavi devono essere create in AWS KMS prima di poter essere utilizzate in ElastiCache.

Per informazioni su come creare chiavi master KMS AWS, consulta [Creazione di chiavi](#) nella AWSGuida per gli sviluppatori di Key Management Service.

ElastiCache consente l'integrazione con AWS KMS. Per ulteriori informazioni, consulta [Utilizzo di concessioni](#) nella AWSGuida per gli sviluppatori Key Management Service. Non è necessaria nessuna operazione per abilitare l'integrazione Amazon ElastiCache con KMSAWS.

La chiave di condizione `kms:ViaService` limita l'uso di unAWS Chiave KMS (chiave KMS) alle richieste da servizi AWS specificati. Per utilizzare `kms:ViaService` con ElastiCache, includere entrambi i nomi di ViaService nella chiave valore di condizione: `elasticache.AWS_region.amazonaws.com` e `dax.AWS_region.amazonaws.com`. Per ulteriori informazioni, consulta [kms:ViaService](#).

È possibile utilizzare [CloudTrailAWS](#) per monitorare le richieste che Amazon ElastiCache invia a AWS Key Management Service per tuo conto. Tutte le chiamate API a AWS Key Management Service riguardo le chiavi gestite dal cliente hanno registri CloudTrail corrispondenti. Puoi anche vedere le concessioni che ElastiCache crea chiamando l'API KMS [ListGrants](#).

- Se elimini o [disabiliti](#) la chiave e [revochi le assegnazioni](#) della chiave utilizzata per crittografare un gruppo di replica, la cache diventa irrecuperabile. In altre parole, non può essere modificato o recuperato dopo un errore hardware. AWS KMS elimina le chiavi master solo dopo un periodo di attesa di almeno sette giorni. Una volta eliminata la chiave, puoi utilizzare una chiave gestita dal cliente differente per creare un backup per scopi di archiviazione.
- La rotazione automatica delle chiavi permette di mantenere le proprietà delle chiavi root KMS AWS e di conseguenza non ha effetto sulla tua capacità di accedere ai dati ElastiCache. Le

cache Amazon ElastiCache crittografate non supportano la rotazione manuale delle chiavi, che comporta la creazione di una nuova chiave root e l'aggiornamento di tutti i riferimenti alla chiave precedente. Per ulteriori informazioni, consulta [RotazioneAWS delle chiavi KMS](#) nella AWSGuida per gli sviluppatori di Key Management Service.

- La crittografia di una cache ElastiCache mediante una chiave KMS richiede una singola assegnazione per ogni cache. Questa assegnazione viene utilizzata per tutta la durata della cache.
- Per ulteriori informazioni su concessioni o limiti KMSAWS, consulta [Limiti](#) nella AWSGuida per gli sviluppatori di Key Management Service.

Abilitazione della crittografia dei dati inattivi

Tutte le cache serverless dispongono della crittografia dei dati a riposo abilitata.

Puoi abilitare la crittografia dei dati a riposo solo durante la creazione di una cache ElastiCache. A tale scopo, è possibile utilizzare il comandoAWS Management Console, ilAWS CLI o l'API ElastiCache.

Durante la creazione di una cache, puoi scegliere una delle opzioni seguenti:

- Predefinita – Questa opzione utilizza la crittografia dei dati inattivi gestita dal servizio.
- Chiave gestita dal cliente – Questa opzione consente di fornire l'ID/ARN della chiave da KMS AWS per la crittografia dei dati inattivi.

Per informazioni su come creare chiavi root KMSAWS, consulta [Creare chiavi](#) nella AWSGuida per gli sviluppatori di Key Management Service

Indice

- [Abilitazione della crittografia dei dati inattivi mediante la AWS Management Console](#)

Abilitazione della crittografia dei dati inattivi mediante la AWS Management Console

Abilitazione della crittografia dei dati a riposo su una cache serverless (console)

Tutte le cache serverless dispongono della crittografia dei dati a riposo abilitata. La chiave KMS di proprietà di AWS viene utilizzata per impostazione predefinita per crittografare i dati. Per scegliere la tua chiave AWS KMS, procedi come segue:

- Espandi la sezione Impostazioni predefinite.

- Scegli Personalizza le impostazioni predefinite nella sezione Impostazioni predefinite.
- Scegli Personalizza le impostazioni di sicurezza nella sezione Sicurezza.
- Scegli CMK gestita dal cliente per l'impostazione Chiave di crittografia.
- Seleziona una chiave nell'impostazione Chiave AWS KMS.

Vedi anche

- [Amazon VPC e sicurezza di ElastiCache](#)
- [Identity and Access Management per Amazon ElastiCache](#)

Riservatezza del traffico Internet

Amazon ElastiCache utilizza le seguenti tecniche per rendere sicuri i dati nella cache e proteggerli dagli accessi non autorizzati:

- [Amazon VPC e sicurezza di ElastiCache](#) spiega il tipo di gruppo di sicurezza necessario per l'installazione.
- [Identity and Access Management per Amazon ElastiCache](#) per concedere e limitare le operazioni di utenti, gruppi e ruoli.

Amazon VPC e sicurezza di ElastiCache

Poiché la sicurezza dei dati è importante, ElastiCache offre i mezzi per controllare chi ha accesso ai tuoi dati. Il modo in cui controlli l'accesso ai tuoi dati dipende dal fatto che tu abbia avviato o meno i cluster in un Virtual Private Cloud (Amazon VPC) di Amazon o Amazon EC2-Classic.

Important

Abbiamo reso obsoleto l'uso di Amazon EC2-Classic per avviare i cluster di ElastiCache. Tutti i nodi della generazione corrente vengono avviati solo in Amazon Virtual Private Cloud.

Il servizio Virtual Private Cloud (Amazon VPC) di Amazon definisce una rete virtuale che ricorda molto un data center tradizionale. Quando vuoi configurare Amazon VPC, puoi selezionare l'intervallo di indirizzi IP, creare sottoreti e configurare tabelle di routing, gateway di rete e le impostazioni di

sicurezza. Puoi anche aggiungere un cluster di cache alla rete virtuale e controllare l'accesso al cluster di cache usando i gruppi di sicurezza di Amazon VPC.

Questa sezione spiega come configurare manualmente un cluster di ElastiCache in un Amazon VPC. Queste informazioni sono rivolte agli utenti che desiderano comprendere come interagiscono ElastiCache e Amazon VPC.

Argomenti

- [Informazioni su ElastiCache e Amazon VPC](#)
- [Modelli di accesso a una cache ElastiCache in un Amazon VPC](#)
- [Creazione di un virtual private cloud \(VPC\).](#)
- [Connessione a una cache in esecuzione in un Amazon VPC](#)

Informazioni su ElastiCache e Amazon VPC

ElastiCache è completamente integrato con il Virtual Private Cloud (Amazon VPC) di Amazon. Per gli utenti di ElastiCache, questo significa che:

- Se l'account AWS supporta solo la piattaforma EC2-VPC, ElastiCache avvia sempre il cluster in un Amazon VPC.
- Se non hai mai utilizzato AWS, i tuoi cluster saranno implementati in un Amazon VPC. Verrà creato automaticamente un VPC di default.
- Se disponi di un VPC predefinito e non specifichi una sottorete all'avvio di un cluster, il cluster si avvia nel Amazon VPC predefinito.

Per ulteriori informazioni, consulta la sezione relativa al [rilevamento delle piattaforme supportate e di un eventuale VPC di default](#).

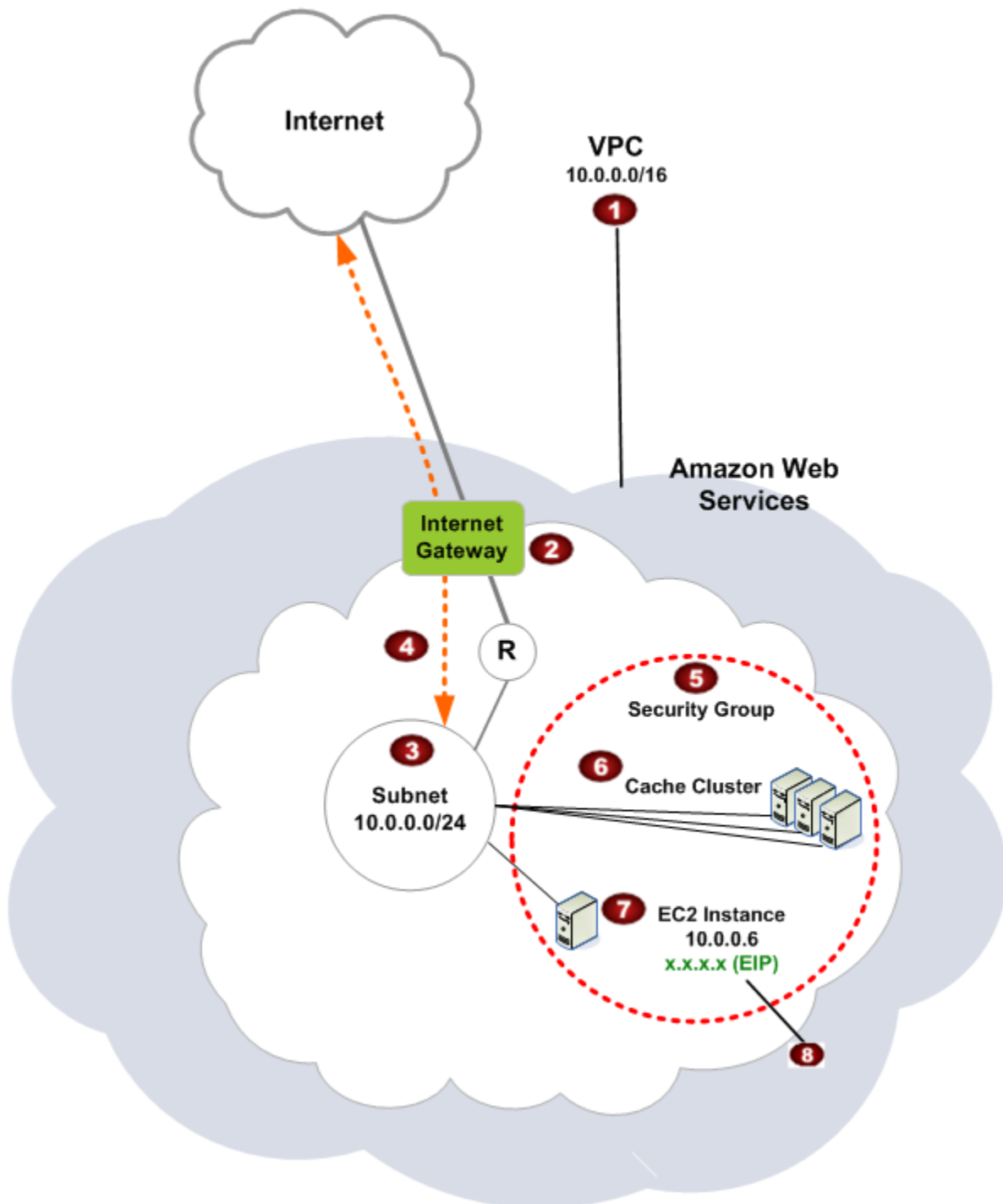
Con Amazon Virtual Private Cloud puoi creare una rete virtuale nel AWS cloud che ricorda molto un data center tradizionale. Puoi configurare il Amazon VPC, inclusi la selezione degli intervalli di indirizzi IP, la creazione di sottoreti, la configurazione di tabelle di routing, i gateway di rete, nonché le impostazioni di sicurezza.

La funzionalità di base di ElastiCache è la stessa di quella di un cloud privato virtuale; ElastiCache gestisce gli aggiornamenti del software, le patch, il rilevamento dell'errore e il ripristino a seconda che i cluster siano implementati all'interno o all'esterno di una Amazon VPC.

I nodi di cache di ElastiCache implementati all'esterno di una Amazon VPC vengono assegnati a un indirizzo IP sul quale si risolve il nome DNS/endpoint. Ciò consente la connettività dalle istanze di Amazon Elastic Compute Cloud (Amazon EC2). Quando avvii un cluster di ElastiCache in una sottorete privata di Amazon VPC, a ogni nodo di cache viene assegnato un indirizzo IP privato all'interno di tale sottorete.

Panoramica di ElastiCache in una Amazon VPC

Il diagramma e la tabella seguenti descrivono l'ambiente di Amazon VPC insieme ai cluster di ElastiCache e alle istanze Amazon EC2 che vengono avviate nella Amazon VPC.

**1**

L'Amazon VPC è una porzione isolata della AWS Cloud a cui è stato assegnato il proprio blocco di indirizzi IP.

2

Un gateway Internet collega la Amazon VPC direttamente a Internet e fornisce accesso alle altre risorse AWS quali Amazon Simple Storage Service (Amazon S3) in esecuzione al di fuori della Amazon VPC.

3

Una sottorete di Amazon VPC è un segmento dell'intervallo di indirizzi IP di una Amazon VPC in cui è possibile isolare le risorse AWS in base alle esigenze operative e di sicurezza.

4

Una tabella di routing nella Amazon VPC instrada il traffico di rete tra la sottorete e Internet. L'Amazon VPC ha un router implicito che, in questo diagramma, è rappresentato dal simbolo del cerchio con la R.

5

Un gruppo di sicurezza di Amazon VPC controlla il traffico in entrata e in uscita per i cluster di ElastiCache e le istanze Amazon EC2.

6

È possibile avviare un cluster di ElastiCache nella sottorete. I nodi della cache hanno indirizzi IP privati dall'intervallo di indirizzi della sottorete.

7

Puoi anche avviare le istanze Amazon EC2 nella sottorete. Ogni istanza Amazon EC2 ha un indirizzo IP privato dall'intervallo di indirizzi della sottorete. L'istanza Amazon EC2 può connettersi a un qualsiasi nodo di cache nella stessa sottorete.

8

Affinché un'istanza Amazon EC2 nella Amazon VPC sia raggiungibile da Internet, è necessario assegnare un indirizzo pubblico, statico, chiamato indirizzo IP elastico all'istanza.

Prerequisiti

Per creare un cluster di ElastiCache all'interno di una Amazon VPC, l'Amazon VPC deve soddisfare i seguenti requisiti:

- L'Amazon VPC deve consentire istanze Amazon EC2 non dedicate. Non puoi utilizzare ElastiCache in una Amazon VPC configurata per una locazione dell'istanza dedicata.

- È necessario definire un gruppo di sottorete di cache per la Amazon VPC. ElastiCache utilizza quel gruppo di sottorete di cache per selezionare una sottorete e gli indirizzi IP all'interno di quella sottorete da associare agli endpoint VPC o ai nodi di cache.
- I blocchi CIDR per ogni sottorete devono essere abbastanza grandi da fornire indirizzi IP di riserva affinché ElastiCache possa utilizzarli durante le attività di manutenzione.

Routing e sicurezza

È possibile configurare il routing nell'Amazon VPC per controllare i punti di concentrazione dei flussi di traffico (ad esempio, il gateway Internet o il gateway virtuale privato). Con un gateway Internet, l'Amazon VPC accede direttamente alle altre risorse AWS che non sono in esecuzione nell'Amazon VPC. Se scegliere di avere solo un gateway privato virtuale con una connessione alla rete locale della tua organizzazione, puoi instradare il traffico Internet attraverso la VPN e utilizzare il firewall e le policy di sicurezza locali per controllare i punti di uscita. In tal caso, incorri in costi aggiuntivi per la larghezza di banda quando accedi alle risorse AWS in Internet.

Puoi utilizzare i gruppi di sicurezza di Amazon VPC per contribuire a proteggere i cluster di ElastiCache e le istanze Amazon EC2 nella Amazon VPC. I gruppi di sicurezza operano come un firewall a livello di istanza, non di sottorete.

Note

Consigliamo di utilizzare i nomi DNS per effettuare il collegamento ai nodi della cache perché l'indirizzo IP sottostante può cambiare.

Documentazione Amazon VPC

Amazon VPC dispone della propria documentazione che descrive come creare e usare l'Amazon VPC. La tabella seguente fornisce i collegamenti alle guide di Amazon VPC.

Descrizione	Documentazione
Come iniziare a usare Amazon VPC	Nozioni di base su Amazon VPC
Come usare Amazon VPC tramite la AWS Management Console	Guida per l'utente di Amazon VPC

Descrizione	Documentazione
Descrizioni complete di tutti i comandi di Amazon VPC	Guida di riferimento alla riga di comando di Amazon EC2 (i comandi di Amazon VPC si trovano nel riferimento Amazon EC2)
Descrizioni complete di operazioni API, tipi di dati ed errori di Amazon VPC	Guida di riferimento all'API di Amazon EC2 (le operazioni dell'API di Amazon VPC si trovano nel riferimento Amazon EC2)
Informazioni per l'amministratore di reti che deve configurare il gateway di una connessione VPN IPsec opzionale presso la sede dell'utente	Cos'è AWS Site-to-Site VPN?

Per informazioni più dettagliate su Amazon Virtual Private Cloud, consulta [Amazon Virtual Private Cloud](#).

Modelli di accesso a una cache ElastiCache in un Amazon VPC

Amazon ElastiCache supporta i seguenti scenari per accedere a una cache in un Amazon VPC:

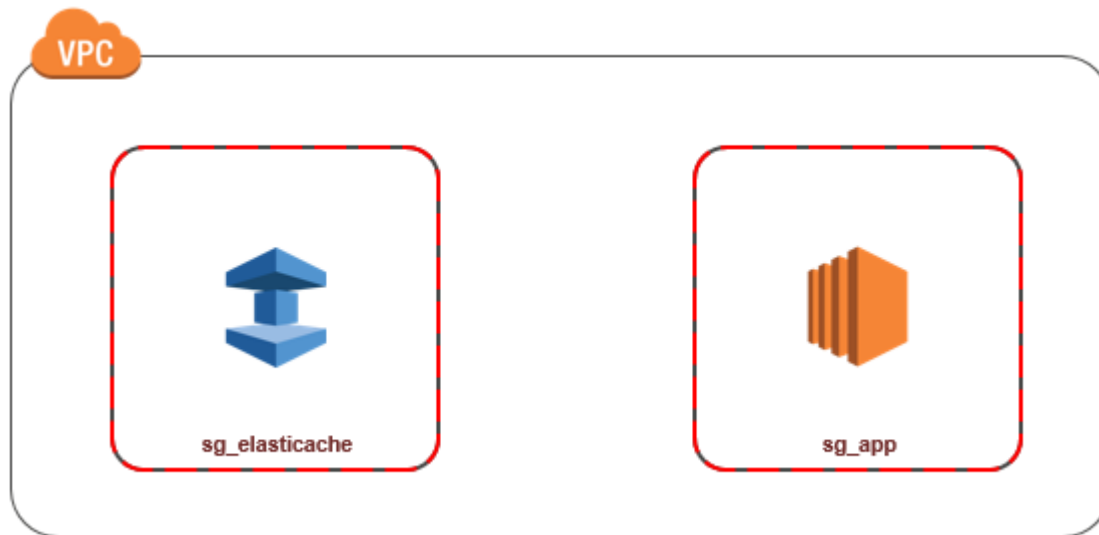
Indice

- [Accesso a una cache ElastiCache quando la cache e l'istanza Amazon EC2 sono nello stesso Amazon VPC](#)
- [Accesso a una cache ElastiCache quando la cache e l'istanza Amazon EC2 sono in differenti Amazon VPC](#)
 - [Accesso a una cache ElastiCache quando la cache e l'istanza Amazon EC2 sono in differenti Amazon VPC nella stessa regione](#)
 - [Uso del Transit Gateway](#)
 - [Accesso a una cache ElastiCache quando la cache e l'istanza Amazon EC2 sono in Amazon VPC differenti in regioni differenti](#)
 - [Utilizzo di VPC di transito](#)
- [Accesso a una cache ElastiCache da un'applicazione in esecuzione nel data center di un cliente](#)
 - [Accesso a una cache ElastiCache da un'applicazione in esecuzione nel data center di un cliente tramite connessione VPN](#)
 - [Accesso a una cache ElastiCache da un'applicazione in esecuzione nel data center di un cliente tramite Direct Connect](#)

Accesso a una cache ElastiCache quando la cache e l'istanza Amazon EC2 sono nello stesso Amazon VPC

Il caso d'uso più comune è quando un'applicazione implementata su un'istanza EC2 deve connettersi a una cache nello stesso VPC.

Nel seguente diagramma viene illustrato questo scenario:



Il modo più semplice per gestire l'accesso tra istanze EC2 e cache nello stesso VPC è procedere come segue:

1. Crea un gruppo di sicurezza VPC per la cache. Questo gruppo di sicurezza può essere usato per limitare l'accesso alla cache. Per questo gruppo di sicurezza è, ad esempio, possibile creare una regola personalizzata che consenta l'accesso TCP tramite la porta assegnata alla cache al momento della creazione e un indirizzo IP che viene utilizzato per accedere alla cache.

La porta predefinita per le cache Memcached è 11211.

2. Creare un gruppo di sicurezza VPC per le istanze EC2 (server Web e di applicazioni). Questo gruppo di sicurezza può, se necessario, consentire l'accesso all'istanza EC2 da Internet tramite la tabella di routing del VPC. Ad esempio, può impostare regole in questo gruppo di sicurezza per consentire l'accesso TCP all'istanza EC2 sulla porta 22.
3. Crea le regole personalizzate nel gruppo di sicurezza per la cache che consentano le connessioni dal gruppo di sicurezza creato per le istanze EC2. In tal modo qualsiasi membro del gruppo di sicurezza può accedere alle cache.

Note

Se si prevede di utilizzare [Local Zones](#), assicurati di averlo abilitato. Quando si crea un gruppo di sottorete in quella zona locale, la VPC viene estesa a quella zona locale e la VPC considera la sottorete come qualsiasi sottorete in qualsiasi altra zona di disponibilità. Tutti i gateway e le tabelle di routing pertinenti verranno regolati automaticamente.

Per creare in un gruppo di sicurezza VPC una regola che consenta connessioni da un altro gruppo di sicurezza

1. Accedere alla Console di gestione AWS e aprire la console di Amazon VPC all'indirizzo <https://console.aws.amazon.com/vpc>.
2. Fare clic su Security Groups (Gruppi di sicurezza) nel riquadro di navigazione.
3. Scegli o crea un gruppo di sicurezza da utilizzare per la cache. In Regole in entrata, scegliere Modifica regole in entrata e quindi Aggiungi regola. Tale gruppo di sicurezza consentirà di accedere ai membri di un altro gruppo di sicurezza.
4. In Type (Tipo) scegliere Custom TCP Rule (Regola TCP personalizzata).
 - a. Per Intervallo di porte specifica la porta utilizzata alla creazione della cache.

La porta predefinita per le cache Memcached è 11211.
 - b. Nella casella Source (fonte) iniziare a digitare l'ID del gruppo di sicurezza. Nell'elenco selezionare il gruppo di sicurezza da utilizzare per le istanze Amazon EC2.
5. Scegliere Save (Salva) al termine.

Type	Protocol	Port Range	Source
Custom TCP Rule	TCP	6379	Custom sg_app

sg-99fc5ce6 - sg_app

Add Rule Cancel Save

Accesso a una cache ElastiCache quando la cache e l'istanza Amazon EC2 sono in differenti Amazon VPC

Quando la cache si trova in un VPC differente da quello in cui si trova l'istanza EC2 che utilizzi per accedervi, è possibile accedere alla cache in diversi modi. Se la cache e l'istanza EC2 si trovano in VPC differenti, ma nella stessa regione, puoi utilizzare il peering VPC. Se la cache e l'istanza EC2 si trovano in regioni differenti, puoi creare una connessione VPN tra le regioni.

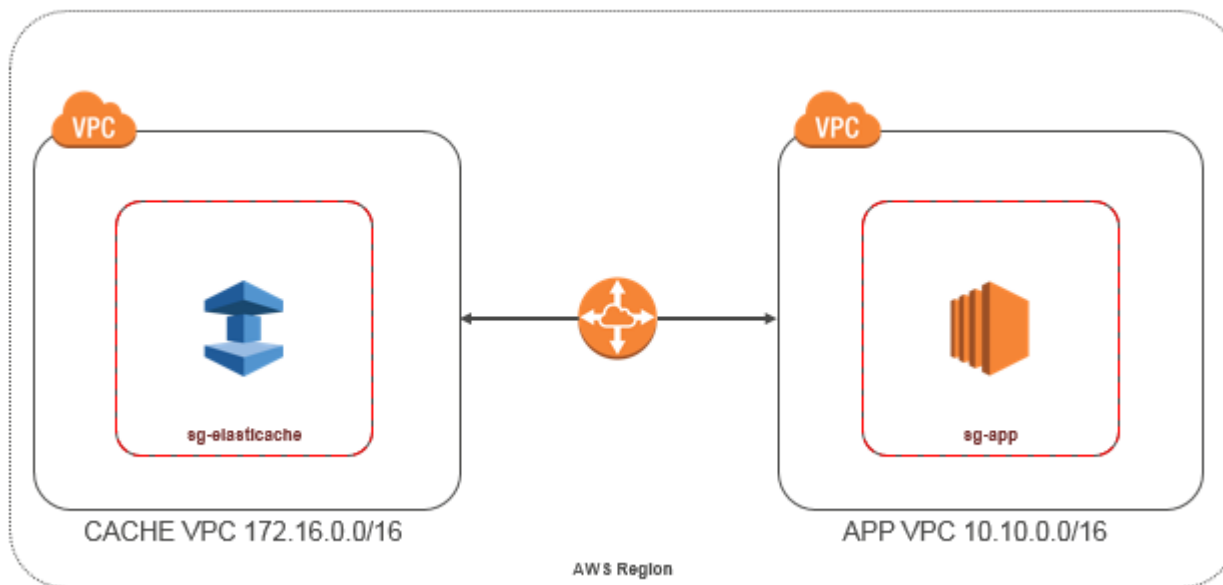
Argomenti

- [Accesso a una cache ElastiCache quando la cache e l'istanza Amazon EC2 sono in differenti Amazon VPC nella stessa regione](#)

- [Accesso a una cache ElastiCache quando la cache e l'istanza Amazon EC2 sono in Amazon VPC differenti in regioni differenti](#)

Accesso a una cache ElastiCache quando la cache e l'istanza Amazon EC2 sono in differenti Amazon VPC nella stessa regione

Il seguente diagramma illustra l'accesso a una cache da un'istanza Amazon EC2 in un Amazon VPC differente presente nella stessa regione, usando una connessione peering Amazon VPC.



Cache a cui si accede da un'istanza Amazon EC2 in un Amazon VPC differente nella stessa regione
- Connessione peering VPC

Una connessione di peering di VPC è una connessione di rete tra due VPC che consentono di instradare il traffico tra loro utilizzando degli indirizzi IP privati. Le istanze in uno qualsiasi dei VPC possono comunicare tra loro come se fossero nella stessa rete. Puoi anche creare una connessione peering VPC tra le tue Amazon VPC o con una Amazon VPC in un altro account AWS, all'interno di una singola regione. Per ulteriori informazioni sul peering VPC di Amazon consulta la [documentazione relativa alla VPC](#).

Note

La risoluzione dei nomi DNS potrebbe non riuscire per i VPC con peering, a seconda delle configurazioni applicate al VPC ElastiCache. Per risolvere tale problema, entrambi i VPC

devono essere abilitati per nomi host DNS e risoluzione DNS. Per ulteriori informazioni, consulta [Abilitazione della risoluzione DNS per una connessione peering VPC](#).

Per accedere a una cache in un Amazon VPC differente con peering

1. Assicurarsi che i due VPC non abbiano un intervallo IP sovrapposto, altrimenti non si potrà sottoporli a peering.
2. Eseguire il peering dei due VPC. Per ulteriori informazioni, consulta [Creare e accettare una connessione peering VPC di Amazon](#).
3. Aggiornare la tabella di routing. Per ulteriori informazioni, consulta [Aggiornamento delle tabelle di routing per una connessione peering VPC](#).

Di seguito è mostrato l'aspetto delle tabelle di routing relative all'esempio nel diagramma precedente. pcx-a894f1c1 è la connessione peering.

Destination	Target	Destination	Target
172.16.0.0/16	local	10.10.0.0/16	local
10.10.0.0/16	pcx-a894f1c1	0.0.0.0/0	igw-bfdcccd8
		172.16.0.0/16	pcx-a894f1c1

Tabella di routing VPC

4. Modifica il gruppo di sicurezza della cache ElastiCache per consentire la connessione in entrata del gruppo di sicurezza dell'applicazione nel VPC in peering. Per ulteriori informazioni, vedi l'argomento relativo ai [gruppi di sicurezza nel VPC in peering](#).

L'accesso a una cache con una connessione peering implica ulteriori costi di trasferimento dei dati.

Uso del Transit Gateway

Un Transit Gateway consente di collegare VPC e connessioni VPN nella stessa regione AWS e routing di traffico tra loro. Un Transit Gateway funziona su tutti gli account AWS ed è possibile utilizzare AWS Resource Access Manager di AWS per condividere il Transit Gateway con altri account. Dopo aver condiviso un gateway di transito con un altro account AWS, il proprietario dell'account può collegare i propri VPC al gateway di transito. Un utente di uno qualsiasi degli account può eliminare il collegamento in qualsiasi momento.

È possibile abilitare il multicast in un gateway di transito, quindi creare un dominio del gateway di transito multicast che consenta l'invio del traffico multicast dall'fonte multicast ai membri del gruppo multicast tramite allegati VPC associati al dominio.

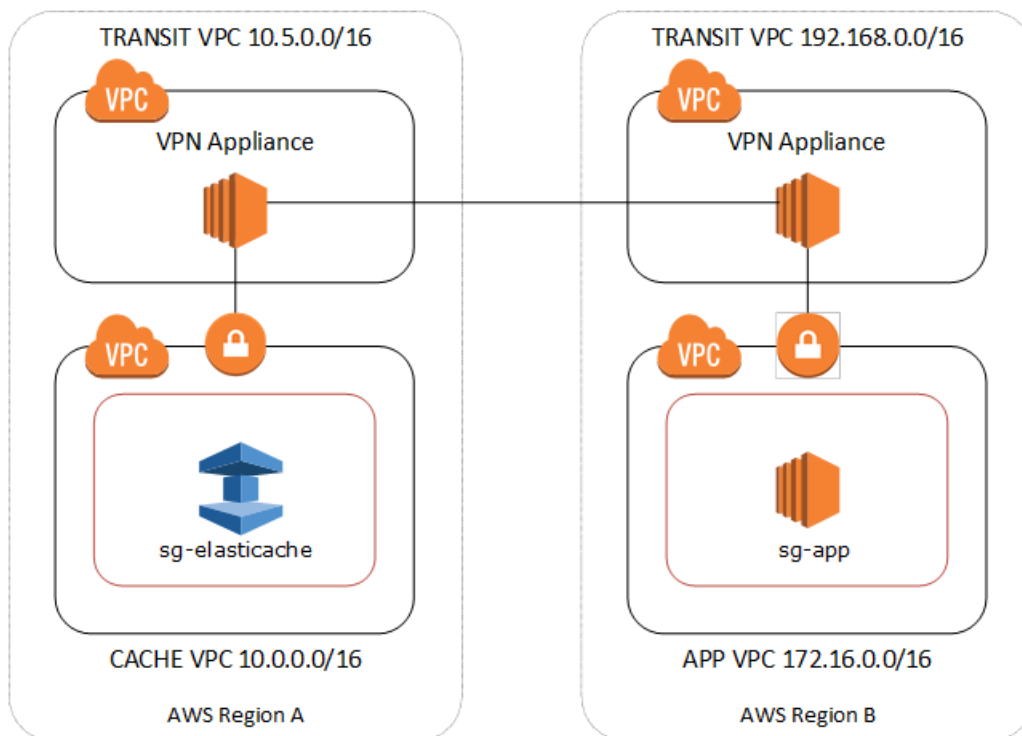
È inoltre possibile creare un allegato di connessione di peering tra gateway di transito in diverse regioni AWS. In questo modo è possibile instradare il traffico tra gli allegati dei gateway di transito in diverse regioni.

Per ulteriori informazioni, consulta [Gateway di transito](#).

Accesso a una cache ElastiCache quando la cache e l'istanza Amazon EC2 sono in Amazon VPC differenti in regioni differenti

Utilizzo di VPC di transito

Un'alternativa all'utilizzo del peering VPC, un'altra strategia comune per connettere più VPC situati in differenti zone geografiche e reti remote consiste nel creare un VPC di transito utilizzabile come centro di transito di rete globale. Un VPC di transito semplifica la gestione della rete e riduce al minimo il numero di connessioni necessarie per connettersi a più VPC e reti remote. Questo tipo di progettazione può consentirti di risparmiare tempo, limitare il lavoro necessario e ridurre i costi, in quanto è praticamente implementato senza la spesa in genere necessaria per stabilire una presenza fisica in un hub di transito di co-location o per distribuire un'apparecchiatura di rete fisica.



Connessione tra VPC differenti in diverse regioni

Una volta che viene definito l'Amazon VPC di transito, un'applicazione implementata in un VPC "spoke" in una regione può essere connessa a una cache ElastiCache in un VPC "spoke" all'interno di un'altra regione.

Per accedere a una cache in un VPC differente in una regione AWS differente

1. Distribuire una soluzione VPC di transito. Per ulteriori informazioni, consulta [AWS Transit Gateway](#).
2. Aggiornare le tabelle di routing VPC nei VPC di app e cache per instradare il traffico tramite il gateway privato virtuale (VGW) e l'appliance VPN. Nel caso di un routing dinamico con Border Gateway Protocol (BGP) le route possono essere automaticamente propagate.
3. Modifica il gruppo di sicurezza della cache ElastiCache per consentire la connessione in entrata dall'intervallo IP delle istanze dell'applicazione. In questo scenario, non è possibile fare riferimento al gruppo di sicurezza del server di applicazioni.

L'accesso a una cache tra regioni introduce latenze di rete e ulteriori costi di trasferimento dei dati tra regioni.

Accesso a una cache ElastiCache da un'applicazione in esecuzione nel data center di un cliente

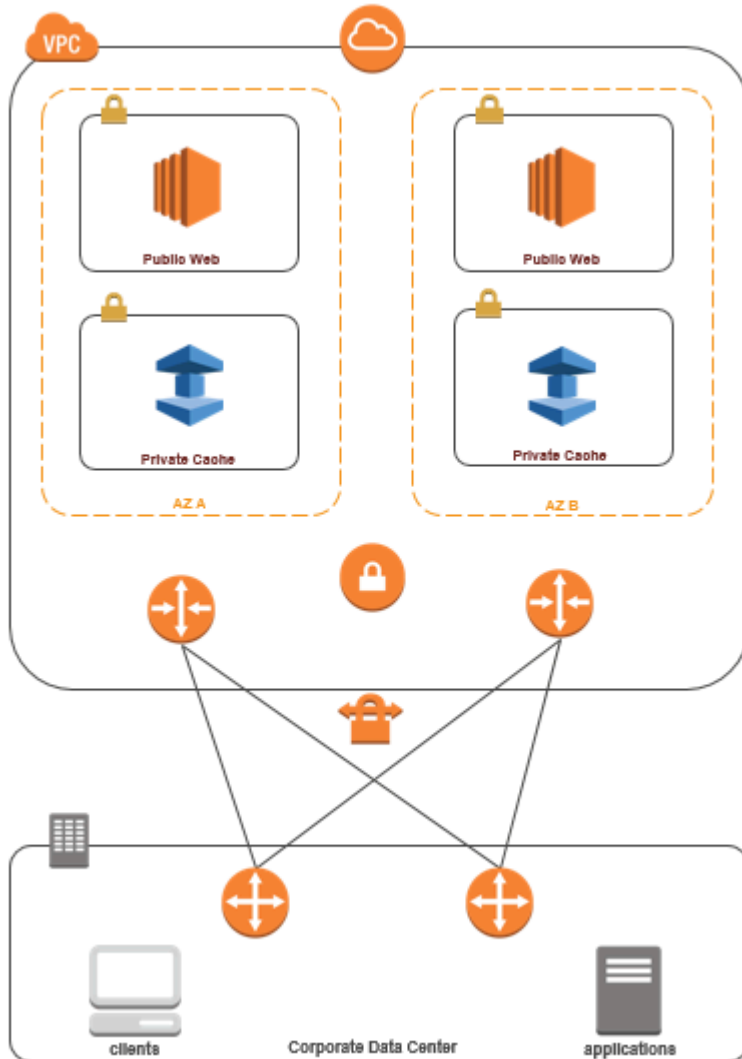
Un altro possibile scenario è rappresentato da un'architettura ibrida in cui i client o le applicazioni nel data center del cliente possono dover accedere a una cache ElastiCache nel VPC. Anche questo scenario è supportato purché sia disponibile una connessione tra VPC e data center dei clienti tramite VPN o Direct Connect.

Argomenti

- [Accesso a una cache ElastiCache da un'applicazione in esecuzione nel data center di un cliente tramite connessione VPN](#)
- [Accesso a una cache ElastiCache da un'applicazione in esecuzione nel data center di un cliente tramite Direct Connect](#)

Accesso a una cache ElastiCache da un'applicazione in esecuzione nel data center di un cliente tramite connessione VPN

Il diagramma seguente illustra l'accesso a una cache ElastiCache da un'applicazione in esecuzione nella rete aziendale tramite connessioni VPN.



Connessione a ElastiCache dal data center tramite una VPN

Per accedere a una cache in un VPC da un'applicazione on-premise con una connessione VPN

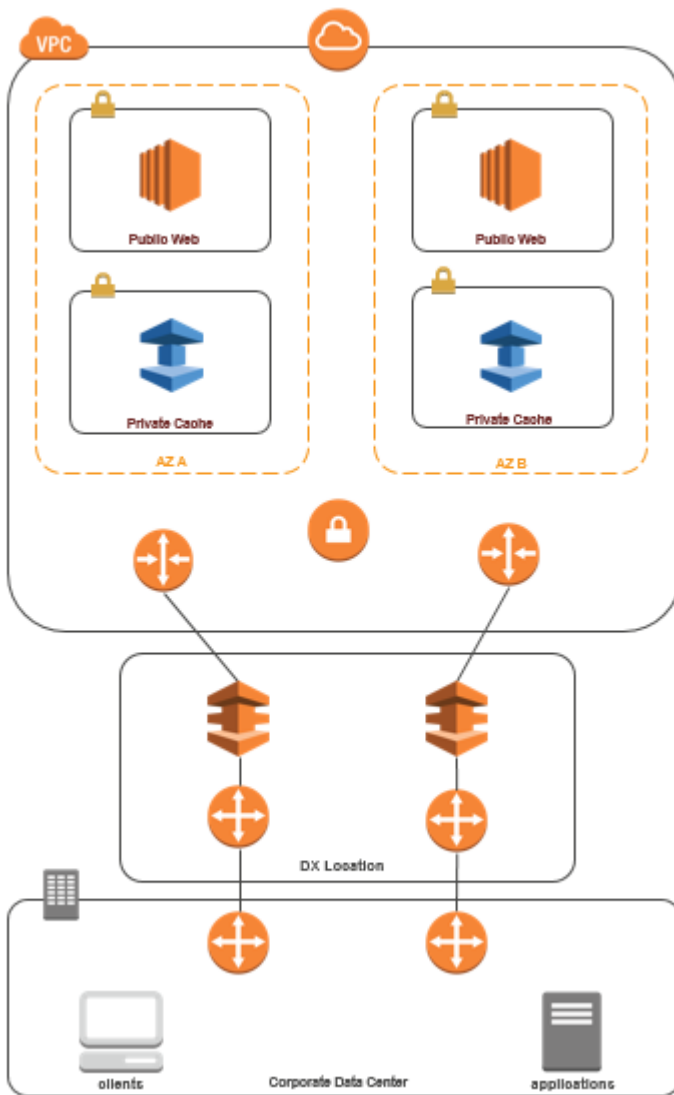
1. Stabilire una connessione VPN aggiungendo un gateway privato virtuale hardware al proprio VPC. Per ulteriori informazioni, consulta [Aggiunta di un gateway privato virtuale hardware al proprio VPC](#).

2. Aggiorna la tabella di routing del VPC per la sottorete in cui viene implementata la cache ElastiCache per consentire il traffico dai server di applicazioni on-premise. Nel caso di un routing dinamico con BGP le route possono essere automaticamente propagate.
3. Modifica il gruppo di sicurezza della cache ElastiCache per consentire la connessione in entrata dai server di applicazioni on-premise.

L'accesso a una cache con una connessione VPN introduce latenze di rete e ulteriori costi di trasferimento dei dati.

Accesso a una cache ElastiCache da un'applicazione in esecuzione nel data center di un cliente tramite Direct Connect

Il diagramma seguente illustra l'accesso a una cache ElastiCache da un'applicazione in esecuzione nella rete aziendale tramite Direct Connect.



Connessione a ElastiCache dal data center tramite Direct Connect

Per accedere a una cache ElastiCache da un'applicazione in esecuzione nella rete tramite Direct Connect

1. Stabilire una connessione Direct Connect. Per ulteriori informazioni, consulta [Nozioni di base su AWS Direct Connect](#).
2. Modifica il gruppo di sicurezza della cache ElastiCache per consentire la connessione in entrata dai server di applicazioni on-premise.

L'accesso a una cache con una connessione DX può introdurre latenze di rete e ulteriori costi di trasferimento dei dati.

Creazione di un virtual private cloud (VPC).

In questo esempio, si crea un Amazon VPC con una sottorete privata per ogni zona di disponibilità.

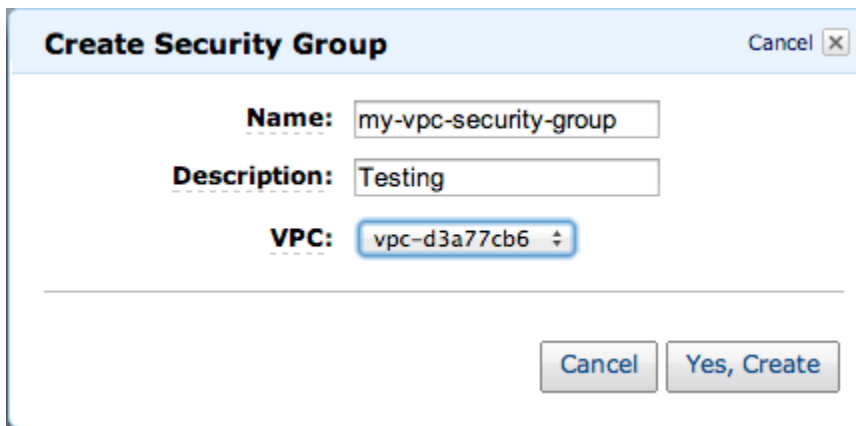
Creazione di un Amazon VPC (Console)

1. Accedere alla Console di gestione AWS e aprire la console Amazon VPC all'indirizzo <https://console.aws.amazon.com/vpc/>.
2. Nel pannello di controllo VPC, scegli Crea VPC.
3. Per Resources to create (Risorse da creare), scegli VPC and more (VPC e altro).
4. Per Number of Availability Zones (AZs) (Numero di zone di disponibilità [AZ]), scegli il numero di zone di disponibilità in cui desideri avviare le sottoreti.
5. Per Number of public subnets (Numero di sottoreti pubbliche), scegli il numero di sottoreti pubbliche che vuoi aggiungere al tuo VPC.
6. Per Number of private subnets (Numero di sottoreti private), scegli il numero di sottoreti private che vuoi aggiungere al tuo VPC.

Tip

Prendi nota degli identificatori di sottorete, specificando quello pubblico e quello privato. Queste informazioni saranno necessarie in seguito quando avvierai i cluster e aggiungerai un'istanza Amazon EC2 al tuo Amazon VPC.

7. Creare un gruppo di sicurezza Amazon VPC Utilizzerai questo gruppo per il cluster di cache e l'istanza Amazon EC2.
 - a. Nel pannello di navigazione della console di gestione di Amazon VPC, scegliere Security Groups (Gruppi di sicurezza).
 - b. Scegli Crea gruppo di sicurezza.
 - c. Digita un nome e una descrizione per il gruppo di sicurezza nelle caselle corrispondenti. Nella casella VPC, seleziona l'identificativo del tuo Amazon VPC.



- d. Dopo aver selezionato tutte le impostazioni che desideri, scegliere Yes, Create (Crea).
8. Definire una regola di ingresso di rete per il gruppo di sicurezza. Questa regola ti consente di collegarti alla tua istanza Amazon EC2 mediante l'SSH (Secure Shell).
- a. Nel pannello di navigazione, fare clic su Security Groups (Gruppi di sicurezza).
 - b. Occorre trovare il gruppo di sicurezza nell'elenco, quindi selezionarlo.
 - c. In Security groups (Gruppi di sicurezza), scegliere la scheda Inbound (In entrata). Nella casella Create a new rule (Crea una nuova regola), scegliere SSH, quindi Add Rule (Aggiungi regola).
 - d. Impostare i seguenti valori per la nuova regola in entrata per consentire l'accesso HTTP:
 - Tipo: HTTP
 - Fonte: 0.0.0.0/0

Scegliere Apply Rule Changes (Applica modifiche della regola).

Ora sei pronto per creare un gruppo di sottorete della cache e per avviare un cluster di cache nel tuo Amazon VPC.

- [Creazione di un gruppo di sottoreti](#)
- [Creazione di un cluster Memcached \(console\)](#).

Connessione a una cache in esecuzione in un Amazon VPC

Questo esempio illustra come avviare un'istanza Amazon EC2 nel Amazon VPC. Puoi quindi accedere a questa istanza e alla cache ElastiCache in esecuzione in Amazon VPC.

Connessione a una cache in esecuzione in un Amazon VPC (console)

In questo esempio, si crea un'istanza Amazon EC2 nell'Amazon VPC. Puoi utilizzare questa istanza Amazon EC2 per connetterti ai nodi di cache in esecuzione nell'Amazon VPC.

Note

Per informazioni sull'uso di Amazon EC2, consulta la [Guida alle operazioni di base di Amazon EC2](#) nella [Documentazione relativa ad Amazon EC2](#).

Per creare un'istanza Amazon EC2 nel Amazon VPC tramite la console di Amazon EC2

1. Accedi a AWS Management Console e apri la console Amazon EC2 all'indirizzo <https://console.aws.amazon.com/ec2/>.
2. Nella console, scegliere Avvia istanza e attenersi alla seguente procedura:
3. Nella pagina scegliere Amazon Machine Image (AMI), scegliere un'AMI Amazon Linux a 64 bit, quindi scegliere Seleziona.
4. Nella pagina Scegliere un tipo di istanza, scegliere 3. Configura l'istanza.
5. Nella pagina Configura dettagli istanza effettuare le seguenti selezioni:
 - a. Nell'elenco Rete scegli il tuo Amazon VPC.
 - b. Nell'elenco Sottoreti scegliere la sottorete pubblica.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Tag Instance 6. Configure Security Group 7. Review

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot Instances to take advantage pricing, assign an access management role to the instance, and more.

Number of instances ⓘ 1

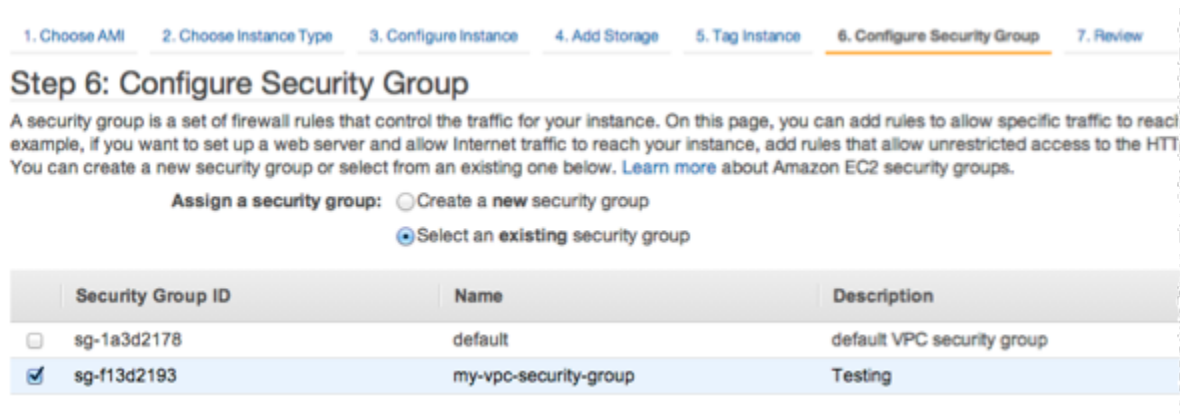
Purchasing option ⓘ Request Spot Instances

Network ⓘ vpc-d3a77cb6 (10.0.0.0/16) : Create new VPC

Subnet ⓘ subnet-58f5e63a(10.0.0.0/24) | sa-east-1a : Create new subnet
250 IP Addresses available

Public IP ⓘ Automatically assign a public IP address to your instances

- Dopo aver selezionato le impostazioni desiderate, scegliere 4. Aggiungi storage.
- Nella pagina Aggiungi archiviazione scegliere 5. Assegna un tag all'istanza.
 - Nella pagina Tag istanza digitare un nome per l'istanza Amazon EC2, quindi scegliere 6. Configura il gruppo di sicurezza.
 - Nella pagina Configura gruppi di sicurezza, seleziona l'opzione Seleziona un gruppo di sicurezza esistente. Per ulteriori informazioni sui gruppi di sicurezza, consulta l'articolo sui [gruppi di sicurezza Amazon EC2 per le istanze Linux](#).



- Scegliere il nome del gruppo di sicurezza di Amazon VPC, quindi scegliere Rivedi e avvia.
- Nella pagina Verifica istanza e avvia, scegliere Avvia.
 - Nella finestra Seleziona una coppia di chiavi esistente o crea una nuova coppia di chiavi, specificare una coppia di chiavi che si desidera utilizzare con questa istanza.

Note

Per informazioni sulla gestione delle coppie di chiavi, consultare la [Guida alle operazioni di base di Amazon EC2](#).

- Quando sei pronto per avviare l'istanza Amazon EC2, scegli Avvia.

È ora possibile assegnare un indirizzo IP elastico all'istanza Amazon EC2 appena creata. È necessario utilizzare questo indirizzo IP per connettersi all'istanza Amazon EC2.

Per assegnare un indirizzo IP elastico (Console)

- Accedere alla console Amazon VPC all'indirizzo <https://console.aws.amazon.com/vpc/>.

2. Nel pannello di navigazione scegliere IP elastici.
3. Scegli Alloca indirizzo IP elastico.
4. Nella finestra di dialogo Alloca indirizzo IP elastico accettare il gruppo di bordi di rete di default e scegliere Alloca .
5. Scegliere l'indirizzo IP elastico appena allocato dall'elenco e scegliere Associa indirizzo.
6. Nella casella di dialogo Associa indirizzo, nella casella Istanza, scegliere l'ID dell'istanza Amazon EC2 che è stata avviata.


Nella casella Indirizzo IP privato scegliere la casella per ottenere l'indirizzo IP privato, quindi scegliere Associa.

È ora possibile utilizzare SSH per connettersi all'istanza Amazon EC2 utilizzando l'indirizzo IP elastico creato.

Eeguire la connessione all'istanza Amazon EC2.

- Aprire una finestra dei comandi. Al prompt dei comandi, eseguire il comando seguente, sostituendo `mykeypair.pem` con il nome del file di coppia di chiavi e `54.207.55.251` con l'indirizzo IP elastico.

```
ssh -i mykeypair.pem ec2-user@54.207.55.251
```

 Important

Non disconnettersi ancora dall'istanza Amazon EC2.

Ora si è pronti per interagire con il cluster di ElastiCache. Prima di farlo, se non lo hai già fatto, devi installare l'utility telnet.

Per installare telnet e interagire con il cluster di cache (AWS CLI)

1. Aprire una finestra dei comandi. Nel prompt dei comandi, eseguire il seguente comando. Alla richiesta di conferma, digitare `y`.

```
sudo yum install telnet  
Loaded plugins: priorities, security, update-motd, upgrade-helper  
Setting up Install Process
```

```
Resolving Dependencies
--> Running transaction check

...(output omitted)...

Total download size: 63 k
Installed size: 109 k
Is this ok [y/N]: y
Downloading Packages:
telnet-0.17-47.7.amzn1.x86_64.rpm                | 63 kB      00:00

...(output omitted)...

Complete!
```

2. Accedere alla console di ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/> e ottenere l'endpoint per uno dei nodi nel cluster di cache. Per maggiori informazioni, consulta [Trovare gli endpoint di connessione](#) per Memcached.
3. Utilizzare telnet per connettersi all'endpoint del nodo della cache sulla porta 11211. Sostituire il nome host mostrato di seguito con il nome host del nodo della cache.

```
telnet my-cache-cluster.7wufxa.0001.use1.cache.amazonaws.com 11211
```

Ora sei connesso al motore della cache e puoi eseguire comandi. In questo esempio, aggiungere un elemento dati alla cache e quindi ottenerlo immediatamente dopo. Infine, disconnettersi dal nodo della cache.

Per memorizzare una chiave e un valore, digitare le due righe seguenti:

```
add mykey 0 3600 28
This is the value for mykey
```

Il motore della cache risponde con quanto segue:

```
OK
```

Per recuperare il valore per mykey, digitare quanto segue:

```
get mykey
```

Il motore della cache risponde con quanto segue:

```
VALUE mykey 0 28  
This is the value for my key  
END
```

Per disconnettersi dal motore della cache, digitare quanto segue:

```
quit
```

Important

Per evitare l'addebito di spese aggiuntive sull'account AWS, assicurarsi di eliminare le risorse AWS non più necessarie dopo aver verificato questi esempi.

API Amazon ElastiCache ed endpoint VPC di interfaccia (AWS PrivateLink)

Puoi stabilire una connessione privata tra gli endpoint VPC e dell'API Amazon ElastiCache creando un endpoint VPC di interfaccia. Gli endpoint di interfaccia sono dotati di tecnologia [AWS PrivateLink](#). AWS PrivateLink ti consente di accedere privatamente alle operazioni dell'API Amazon ElastiCache senza un gateway Internet, un dispositivo NAT, una connessione VPN o una connessione AWS Direct Connect.

Le istanze nel VPC non richiedono indirizzi IP pubblici per la comunicazione con gli endpoint dell'API Amazon ElastiCache. Le istanze non richiedono indirizzi IP pubblici neppure per l'uso di una delle operazioni dell'API ElasticCache disponibili. Il traffico di rete tra VPC e Amazon ElastiCache non esce dalla rete Amazon. Ogni endpoint dell'interfaccia è rappresentato da una o più interfacce di rete elastiche nelle sottoreti. Per maggiori informazioni sulle interfacce di rete elastiche, consulta le [interfacce di rete elastiche](#) nella Guida dell'utente di Amazon EC2.

- Per ulteriori informazioni sugli endpoint dei VPC, consulta [Endpoint VPC di interfaccia \(AWS PrivateLink\)](#) nella Guida per l'utente di Amazon VPC.

- Per ulteriori informazioni sulle operazioni dell'API ElastiCache, consulta [Operazioni dell'API ElastiCache](#).

Dopo aver creato un endpoint VPC di interfaccia, se abiliti nomi host [DNS privato](#) per l'endpoint, per il VPC viene risolto l'endpoint ElastiCache predefinito (<https://elasticache.Region.amazonaws.com>). Se non abiliti nomi host DNS privati, Amazon VPC fornisce un nome di endpoint DNS che puoi utilizzare nel formato seguente:

```
VPC_Endpoint_ID.elasticache.Region.vpce.amazonaws.com
```

Per ulteriori informazioni, consulta [Endpoint VPC di interfaccia \(AWS PrivateLink\)](#) nella Guida per l'utente di Amazon VPC. ElastiCache supporta l'esecuzione di chiamate a tutte le relative [Operazioni dell'API](#) nel VPC.

Note

I nomi host DNS privati possono essere abilitati per un solo endpoint VPC nel VPC. Per creare un endpoint VPC supplementare, il nome host DNS privato deve essere disabilitato.

Considerazioni sugli endpoint VPC

Prima di configurare un endpoint VPC di interfaccia per gli endpoint dell'API Amazon ElastiCache, rivedi [Proprietà e limitazioni degli endpoint di interfaccia](#) nella Guida per l'utente di Amazon VPC. Tutte le operazioni dell'API ElastiCache pertinenti alla gestione delle risorse di Amazon ElastiCache sono disponibili dal VPC mediante AWS PrivateLink.

Le policy degli endpoint VPC sono supportate per gli endpoint dell'API ElastiCache. Per impostazione predefinita, l'accesso completo alle operazioni dell'API ElastiCache è consentito tramite l'endpoint. Per ulteriori informazioni, consultare [Controllo degli accessi ai servizi con endpoint VPC](#) in Guida per l'utente di Amazon VPC.

Creazione di un endpoint VPC di interfaccia per l'API ElastiCache

Puoi creare un endpoint VPC per l'API Amazon ElastiCache tramite la console Amazon VPC o AWS CLI. Per ulteriori informazioni, consulta [Creazione di un endpoint di interfaccia](#) nella Guida per l'utente di Amazon VPC.

Una volta creato un endpoint VPC di interfaccia, puoi abilitare nomi host DNS privati per l'endpoint. Con questa operazione, per l'endpoint VPC viene risolto l'endpoint Amazon ElastiCache predefinito (<https://elasticache.Region.amazonaws.com>). Per le regioni AWS Cina (Pechino) e Cina (Ningxia), puoi creare richieste API con l'endpoint VPC utilizzando `elasticache.cn-north-1.amazonaws.com.cn` per Pechino e `elasticache.cn-northwest-1.amazonaws.com.cn` per Ningxia. Per ulteriori informazioni, consulta [Accesso a un servizio tramite un endpoint di interfaccia](#) in Guida per l'utente di Amazon VPC.

Creazione di una policy degli endpoint VPC per l'API Amazon ElastiCache

Puoi allegare una policy degli endpoint all'endpoint VPC che controlla l'accesso all'API ElastiCache. La policy specifica quanto segue:

- Il principale che può eseguire operazioni.
- Le operazioni che possono essere eseguite.
- Le risorse sui cui si possono eseguire operazioni.

Per ulteriori informazioni, consulta [Controllo degli accessi ai servizi con endpoint VPC](#) in Guida per l'utente di Amazon VPC.

Example Policy degli endpoint VPC per le operazioni dell'API ElastiCache

Di seguito è riportato un esempio di una policy degli endpoint per l'API ElastiCache. Se collegata a un endpoint, questa policy concede l'accesso alle operazioni dell'API ElastiCache elencate per tutti i principali su tutte le risorse.

```
{
  "Statement": [{
    "Principal": "*",
    "Effect": "Allow",
    "Action": [
      "elasticache:CreateCacheCluster",
      "elasticache:ModifyCacheCluster"
    ],
    "Resource": "*"
  }]
}
```

Example Policy degli endpoint VPC che nega tutti gli accessi da un account AWS specificato

La seguente policy di endpoint VPC nega all'account AWS **123456789012** l'accesso completo alle risorse utilizzando l'endpoint. La policy consente tutte le operazioni da altri account.

```
{
  "Statement": [{
    "Action": "*",
    "Effect": "Allow",
    "Resource": "*",
    "Principal": "*"
  },
  {
    "Action": "*",
    "Effect": "Deny",
    "Resource": "*",
    "Principal": {
      "AWS": [
        "123456789012"
      ]
    }
  }
]
```

Sottoreti e gruppi di sottoreti

Un gruppo di sottoreti è una raccolta di sottoreti (generalmente private) che è possibile designare per i cluster progettati autonomamente in esecuzione in un ambiente Amazon Virtual Private Cloud (Amazon VPC).

Se crei un cluster progettato autonomamente in un Amazon VPC, devi usare un gruppo di sottoreti. ElastiCache utilizza il gruppo di sottoreti per selezionare una sottorete e gli indirizzi IP all'interno di quella sottorete da associare ai nodi.

ElastiCache fornisce un gruppo di sottoreti IPv4 predefinito; in alternativa, puoi scegliere di crearne uno nuovo. Per IPv6 devi creare un gruppo di sottoreti con un blocco CIDR IPv6. Se scegli dual-stack, devi selezionare un tipo di individuazione IP: IPv6 o IPv4.

ElastiCache serverless non utilizza una risorsa gruppo di sottoreti e usa invece un elenco di sottoreti direttamente durante la creazione.

Questa sezione specifica come creare e sfruttare le sottoreti e i gruppi di sottoreti per gestire gli accessi alle risorse di ElastiCache.

Per ulteriori informazioni sull'utilizzo dei gruppi di sottoreti in un ambiente Amazon VPC, consulta [Accesso al cluster](#) .

Argomenti

- [Creazione di un gruppo di sottoreti](#)
- [Assegnazione di un gruppo di sottoreti ad una cache](#)
- [Modifica di un gruppo di sottoreti](#)
- [Eliminazione di un gruppo di sottoreti](#)

Creazione di un gruppo di sottoreti

Un gruppo di sottoreti di cache è una raccolta di sottoreti che è possibile assegnare alle cache di un VPC. Quando si avvia una cache in un VPC, è necessario scegliere un gruppo di sottoreti di cache. Quindi, ElastiCache utilizza il gruppo di sottoreti di cache per assegnare gli indirizzi IP della sottorete per ogni nodo di cache.

Quando si crea un nuovo gruppo di sottoreti, tieni presente il numero di indirizzi IP disponibili. Se la sottorete ha un numero molto ridotto di indirizzi IP liberi, potresti avere delle limitazioni sul numero di nodi che puoi aggiungere a un cluster. Per risolvere questo problema, è possibile assegnare una o più sottoreti a un gruppo di sottoreti in modo da avere un numero sufficiente di indirizzi IP nella zona di disponibilità del cluster. Dopodiché, è possibile aggiungere ulteriori nodi al cluster.

Se si sceglie IPV4 come tipo di rete, sarà disponibile un gruppo di sottoreti predefinito; in alternativa, è possibile crearne uno nuovo. ElastiCache utilizza il gruppo di sottoreti per selezionare una sottorete e gli indirizzi IP all'interno di quella sottorete da associare ai nodi. Se si sceglie dual-stack o IPV6, si viene indirizzati alla creazione di sottoreti dual-stack o IPV6. Per ulteriori informazioni sui tipi di rete, consulta [Network type](#). Per ulteriori informazioni, consulta [Creazione di una sottorete nel VPC](#).

Le procedure seguenti illustrano come creare un gruppo di sottoreti chiamato `mysubnetgroup` (console), la AWS CLI e l'API di ElastiCache.

Creazione di un gruppo di sottoreti (Console)

La procedura seguente mostra come creare un gruppo di sottoreti (console).

Per creare un gruppo di sottoreti (Console)

1. Accedere alla Console di gestione di AWS e aprire la console di ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nell'elenco di navigazione scegli Gruppi di sottoreti.
3. Scegli Crea gruppo di sottoreti.
4. Nella procedura guidata Crea gruppo di sottoreti procedi come indicato di seguito. Dopo aver selezionato tutte le impostazioni desiderate, scegli Crea.
 - a. Nella casella Name (Nome), digitare un nome per il gruppo di sottoreti.
 - b. Nella casella Description (Descrizione), digitare una descrizione per il gruppo di sottoreti.
 - c. Nella casella ID VPC seleziona il tuo Amazon VPC.

- d. Per impostazione predefinita sono selezionate tutte le sottoreti. Nel pannello Sottoreti selezionate fai clic su Gestisci e seleziona le zone di disponibilità o le [zone locali](#) e gli ID delle sottoreti private, quindi seleziona Scegli.
5. Nel messaggio di conferma visualizzato, scegliere Close (Chiudi).

Il nuovo gruppo di sottoreti viene visualizzato nell'elenco Subnet Groups (Gruppi di sottoreti) della console di ElastiCache. Nella parte in basso della finestra puoi scegliere il gruppo di sottoreti per visualizzare i dettagli, ad esempio tutte le sottoreti associate a tale gruppo.

Creazione di un gruppo di sottoreti (AWS CLI)

Al prompt dei comandi, utilizzare il comando `create-cache-subnet-group` per creare un gruppo di sottoreti.

Per Linux, macOS o Unix:

```
aws elasticache create-cache-subnet-group \  
  --cache-subnet-group-name mysubnetgroup \  
  --cache-subnet-group-description "Testing" \  
  --subnet-ids subnet-53df9c3a
```

Per Windows:

```
aws elasticache create-cache-subnet-group ^  
  --cache-subnet-group-name mysubnetgroup ^  
  --cache-subnet-group-description "Testing" ^  
  --subnet-ids subnet-53df9c3a
```

Questo comando dovrebbe generare un output simile al seguente:

```
{  
  "CacheSubnetGroup": {  
    "VpcId": "vpc-37c3cd17",  
    "CacheSubnetGroupDescription": "Testing",  
    "Subnets": [  
      {  
        "SubnetIdentifier": "subnet-53df9c3a",  
        "SubnetAvailabilityZone": {  
          "Name": "us-west-2a"  
        }  
      }  
    ]  
  }  
}
```

```
    }  
  ],  
  "CacheSubnetGroupName": "mysubnetgroup"  
}  
}
```

Per ulteriori informazioni, consulta l'argomento della AWS CLI [create-cache-subnet-group](#).

Assegnazione di un gruppo di sottoreti ad una cache

Dopo aver creato un gruppo di sottoreti, puoi avviare una cache in un Amazon VPC. Per ulteriori informazioni, consulta gli argomenti seguenti.

- Cluster Memcached: per avviare un cluster Memcached, consultare [Creazione di un cluster Memcached \(console\)](#). Nel passaggio 7.a (Impostazioni Memcached avanzate), selezionare un gruppo di sottoreti VPC.

Modifica di un gruppo di sottoreti

È possibile modificare una descrizione del gruppo di sottoreti o modificare l'elenco di ID di sottoreti associato al gruppo di sottoreti. Non è possibile eliminare da un gruppo un ID di sottorete se una cache utilizza attualmente quella sottorete.

Le procedure seguenti mostrano come modificare un gruppo di sottoreti.

Modifica dei gruppi di sottoreti (Console)

Per modificare un gruppo di sottoreti

1. Accedere alla AWS Management Console e aprire la console di ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nel pannello di navigazione selezionare Subnet groups (Gruppi di sottoreti).
3. Nell'elenco dei gruppi di sottoreti scegli il pulsante di opzione corrispondente alla sottorete che vuoi modificare e seleziona Modifica.
4. Nel pannello Sottoreti selezionate scegli Gestisci.
5. Apporta eventuali modifiche alle sottoreti selezionate e fai clic su Scegli.
6. Fai clic su Salva le modifiche per salvare le modifiche.

Modifica di gruppi di sottoreti (AWS CLI)

Al prompt dei comandi, utilizzare il comando `modify-cache-subnet-group` per modificare un gruppo di sottoreti.

Per Linux, macOS o Unix:

```
aws elasticache modify-cache-subnet-group \  
  --cache-subnet-group-name mysubnetgroup \  
  --cache-subnet-group-description "New description" \  
  --subnet-ids "subnet-42df9c3a" "subnet-48fc21a9"
```

Per Windows:

```
aws elasticache modify-cache-subnet-group ^  
  --cache-subnet-group-name mysubnetgroup ^  
  --cache-subnet-group-description "New description" ^  
  --subnet-ids "subnet-42df9c3a" "subnet-48fc21a9"
```

Questo comando dovrebbe generare un output simile al seguente:

```
{
  "CacheSubnetGroup": {
    "VpcId": "vpc-73cd3c17",
    "CacheSubnetGroupDescription": "New description",
    "Subnets": [
      {
        "SubnetIdentifier": "subnet-42dcf93a",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2a"
        }
      },
      {
        "SubnetIdentifier": "subnet-48fc12a9",
        "SubnetAvailabilityZone": {
          "Name": "us-west-2a"
        }
      }
    ],
    "CacheSubnetGroupName": "mysubnetgroup"
  }
}
```

Per ulteriori informazioni, consulta l'argomento della AWS CLI [modify-cache-subnet-group](#).

Eliminazione di un gruppo di sottoreti

Se ritieni che il gruppo di sottoreti non sia più necessario, puoi eliminarlo. Non è possibile eliminare un gruppo di sottoreti se è attualmente utilizzato da una cache.

Le procedure seguenti mostrano come eliminare un gruppo di sottoreti.

Eliminazione di un gruppo di sottoreti (Console)

Per eliminare un gruppo di sottoreti

1. Accedere alla AWS Management Console e aprire la console di ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Nel pannello di navigazione selezionare Subnet groups (Gruppi di sottoreti).
3. Nell'elenco dei gruppi di sottoreti, scegliere quello da eliminare, quindi scegliere Delete (Elimina).
4. Quando ti viene chiesto di confermare questa operazione, digita il nome del gruppo di sottoreti nel campo di immissione testo e scegli Elimina.

Eliminazione di un gruppo di sottoreti (AWS CLI)

Utilizzando l'AWS CLI, chiamare il comando `delete-cache-subnet-group` con il seguente parametro:

- `--cache-subnet-group-name mysubnetgroup`

Per Linux, macOS o Unix:

```
aws elasticache delete-cache-subnet-group \  
  --cache-subnet-group-name mysubnetgroup
```

Per Windows:

```
aws elasticache delete-cache-subnet-group ^  
  --cache-subnet-group-name mysubnetgroup
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta l'argomento della AWS CLI [delete-cache-subnet-group](#).

Identity and Access Management per Amazon ElastiCache

AWS Identity and Access Management (IAM) è un Servizio AWS che consente agli amministratori di controllare in modo sicuro l'accesso alle risorse AWS. Gli amministratori IAM controllano chi può essere autenticato (chi ha effettuato l'accesso) e autorizzato (chi dispone di autorizzazioni) a utilizzare le risorse ElastiCache. IAM è un Servizio AWS il cui uso non comporta costi aggiuntivi.

Argomenti

- [Destinatari](#)
- [Autenticazione con identità](#)
- [Gestione dell'accesso con policy](#)
- [Funzionamento di Amazon ElastiCache con IAM](#)
- [Esempi di policy basate su identità per Amazon ElastiCache](#)
- [Risoluzione dei problemi relativi all'identità e all'accesso di Amazon ElastiCache](#)
- [Controllo accessi](#)
- [Panoramica della gestione delle autorizzazioni di accesso alle risorse di ElastiCache](#)

Destinatari

Le modalità di utilizzo di AWS Identity and Access Management (IAM) cambiano in base alle operazioni eseguite in ElastiCache.

Utente del servizio: se si utilizza il servizio ElastiCache per eseguire il lavoro, l'amministratore fornisce le credenziali e le autorizzazioni necessarie. All'aumentare del numero più funzionalità ElastiCache utilizzate per eseguire lavoro, potrebbero essere richieste autorizzazioni aggiuntive. La comprensione della gestione dell'accesso ti consente di richiedere le autorizzazioni corrette all'amministratore. Se non riesci ad accedere a una funzionalità di ElastiCache, consultare [Risoluzione dei problemi relativi all'identità e all'accesso di Amazon ElastiCache](#).

Amministratore del servizio: l'utente responsabile delle risorse ElastiCache presso l'azienda dispone dell'accesso completo a ElastiCache. Il suo compito è determinare le caratteristiche e le risorse ElastiCache a cui gli utenti del servizio devono accedere. Devi inviare le richieste all'amministratore IAM per cambiare le autorizzazioni degli utenti del servizio. Esamina le informazioni contenute in questa pagina per comprendere i concetti di base relativi a IAM. Per ulteriori informazioni su come

l'azienda può utilizzare IAM con ElastiCache, consultare [Funzionamento di Amazon ElastiCache con IAM](#).

Amministratore IAM: in qualità di amministratore IAM, potrebbe essere necessario conoscere i dettagli su come scrivere le policy di gestione dell'accesso a ElastiCache. Per visualizzare policy basate su identità di ElastiCache di esempio che puoi utilizzare in IAM, consultare [Esempi di policy basate su identità per Amazon ElastiCache](#).

Autenticazione con identità

L'autenticazione è la procedura di accesso ad AWS con le credenziali di identità. Devi essere autenticato (connesso a AWS) come utente root Utente root dell'account AWS, come utente IAM o assumere un ruolo IAM.

Puoi accedere ad AWS come identità federata utilizzando le credenziali fornite attraverso un'origine di identità. Gli utenti AWS IAM Identity Center (Centro identità IAM), l'autenticazione Single Sign-On (SSO) dell'azienda e le credenziali di Google o Facebook sono esempi di identità federate. Se accedi come identità federata, l'amministratore ha configurato in precedenza la federazione delle identità utilizzando i ruoli IAM. Se accedi ad AWS tramite la federazione, assumi indirettamente un ruolo.

A seconda del tipo di utente, puoi accedere alla AWS Management Console o al portale di accesso AWS. Per ulteriori informazioni sull'accesso ad AWS, consulta la sezione [Come accedere al tuo Account AWS](#) nella Guida per l'utente di Accedi ad AWS.

Se accedi ad AWS in modo programmatico, AWS fornisce un Software Development Kit (SDK) e un'interfaccia a riga di comando (CLI) per firmare crittograficamente le richieste utilizzando le tue credenziali. Se non utilizzi gli strumenti AWS, devi firmare le richieste personalmente. Per ulteriori informazioni sulla firma delle richieste, consulta [Firma delle richieste AWS](#) nella Guida per l'utente IAM.

A prescindere dal metodo di autenticazione utilizzato, potrebbe essere necessario specificare ulteriori informazioni sulla sicurezza. AWS consiglia ad esempio di utilizzare l'autenticazione a più fattori (MFA) per aumentare la sicurezza dell'account. Per ulteriori informazioni, consulta [Autenticazione a più fattori](#) nella Guida per l'utente di AWS IAM Identity Center e [Utilizzo dell'autenticazione a più fattori \(MFA\) in AWS](#) nella Guida per l'utente di IAM.

Utente root di un Account AWS

Quando crei un Account AWS, inizi con una singola identità di accesso che ha accesso completo a tutti i Servizi AWS e le risorse nell'account. Tale identità è detta utente root Account AWS ed è

possibile accedervi con l'indirizzo e-mail e la password utilizzati per creare l'account. Si consiglia vivamente di non utilizzare l'utente root per le attività quotidiane. Conserva le credenziali dell'utente root e utilizzarle per eseguire le operazioni che solo l'utente root può eseguire. Per un elenco completo delle attività che richiedono l'accesso come utente root, consulta la sezione [Attività che richiedono le credenziali dell'utente root](#) nella Guida per l'utente di IAM.

Identità federata

Come best practice, richiedi agli utenti umani, compresi quelli che richiedono l'accesso di amministratore, di utilizzare la federazione con un provider di identità per accedere a Servizi AWS utilizzando credenziali temporanee.

Un'identità federata è un utente della directory degli utenti aziendali, un provider di identità Web, AWS Directory Service, la directory Identity Center o qualsiasi utente che accede ai Servizi AWS utilizzando le credenziali fornite tramite un'origine di identità. Quando le identità federate accedono agli Account AWS, assumono ruoli e i ruoli forniscono credenziali temporanee.

Per la gestione centralizzata degli accessi, consigliamo di utilizzare AWS IAM Identity Center. È possibile creare utenti e gruppi in IAM Identity Center oppure connettersi e sincronizzarsi con un gruppo di utenti e gruppi nell'origine di identità per utilizzarli in tutte le applicazioni e gli Account AWS. Per ulteriori informazioni sul Centro identità IAM, consulta [Cos'è Centro identità IAM?](#) nella Guida per l'utente di AWS IAM Identity Center.

Utenti e gruppi IAM

Un [utente IAM](#) è una identità all'interno del tuo Account AWS che dispone di autorizzazioni specifiche per una singola persona o applicazione. Ove possibile, consigliamo di fare affidamento a credenziali temporanee invece di creare utenti IAM con credenziali a lungo termine come le password e le chiavi di accesso. Tuttavia, per casi d'uso specifici che richiedono credenziali a lungo termine con utenti IAM, si consiglia di ruotare le chiavi di accesso. Per ulteriori informazioni, consulta la pagina [Rotazione periodica delle chiavi di accesso per casi d'uso che richiedono credenziali a lungo termine](#) nella Guida per l'utente di IAM.

Un [gruppo IAM](#) è un'identità che specifica un insieme di utenti IAM. Non è possibile eseguire l'accesso come gruppo. È possibile utilizzare gruppi per specificare le autorizzazioni per più utenti alla volta. I gruppi semplificano la gestione delle autorizzazioni per set di utenti di grandi dimensioni. Ad esempio, è possibile avere un gruppo denominato Amministratori IAM e concedere a tale gruppo le autorizzazioni per amministrare le risorse IAM.

Gli utenti sono diversi dai ruoli. Un utente è associato in modo univoco a una persona o un'applicazione, mentre un ruolo è destinato a essere assunto da chiunque ne abbia bisogno. Gli utenti dispongono di credenziali a lungo termine permanenti, mentre i ruoli forniscono credenziali temporanee. Per ulteriori informazioni, consulta [Quando creare un utente IAM \(invece di un ruolo\)](#) nella Guida per l'utente di IAM.

Ruoli IAM

Un [ruolo IAM](#) è un'identità all'interno di un Account AWS che dispone di autorizzazioni specifiche. È simile a un utente IAM, ma non è associato a una persona specifica. È possibile assumere temporaneamente un ruolo IAM nella AWS Management Console mediante lo [scambio di ruoli](#). È possibile assumere un ruolo chiamando un'azione AWS CLI o API AWS oppure utilizzando un URL personalizzato. Per ulteriori informazioni sui metodi per l'utilizzo dei ruoli, consulta [Utilizzo di ruoli IAM](#) nella Guida per l'utente di IAM.

I ruoli IAM con credenziali temporanee sono utili nelle seguenti situazioni:

- **Accesso utente federato:** per assegnare le autorizzazioni a una identità federata, è possibile creare un ruolo e definire le autorizzazioni per il ruolo. Quando un'identità federata viene autenticata, l'identità viene associata al ruolo e ottiene le autorizzazioni da esso definite. Per ulteriori informazioni sulla federazione dei ruoli, consulta [Creazione di un ruolo per un provider di identità di terza parte](#) nella Guida per l'utente di IAM. Se utilizzi IAM Identity Center, configura un set di autorizzazioni. IAM Identity Center mette in correlazione il set di autorizzazioni con un ruolo in IAM per controllare a cosa possono accedere le identità dopo l'autenticazione. Per ulteriori informazioni sui set di autorizzazioni, consulta [Set di autorizzazioni](#) nella Guida per l'utente di AWS IAM Identity Center.
- **Autorizzazioni utente IAM temporanee:** un utente IAM o un ruolo può assumere un ruolo IAM per ottenere temporaneamente autorizzazioni diverse per un'attività specifica.
- **Accesso multi-account:** è possibile utilizzare un ruolo IAM per permettere a un utente (un principale affidabile) con un account diverso di accedere alle risorse nell'account. I ruoli sono lo strumento principale per concedere l'accesso multi-account. Tuttavia, per alcuni dei Servizi AWS, è possibile collegare una policy direttamente a una risorsa (anziché utilizzare un ruolo come proxy). Per informazioni sulle differenze tra ruoli e policy basate su risorse per l'accesso multi-account, consulta [Differenza tra i ruoli IAM e le policy basate su risorse](#) nella Guida per l'utente di IAM.
- **Accesso multi-servizio:** alcuni Servizi AWS utilizzano funzionalità in altri Servizi AWS. Ad esempio, quando effettui una chiamata in un servizio, è comune che tale servizio esegua applicazioni in Amazon EC2 o archivi oggetti in Amazon S3. Un servizio può eseguire questa operazione

utilizzando le autorizzazioni dell'entità chiamante, utilizzando un ruolo di servizio o utilizzando un ruolo collegato al servizio.

- Inoltro delle sessioni di accesso (FAS): quando si utilizza un utente o un ruolo IAM per eseguire operazioni in AWS, tale utente o ruolo viene considerato un principale. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra azione in un servizio diverso. FAS utilizza le autorizzazioni del principale che effettua la chiamata a un Servizio AWS, combinate con il Servizio AWS richiedente, per effettuare richieste a servizi a valle. Le richieste FAS vengono effettuate solo quando un servizio riceve una richiesta che necessita di interazioni con altri Servizi AWS o risorse per essere portata a termine. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le operazioni. Per i dettagli delle policy relative alle richieste FAS, consulta la pagina [Forward access sessions](#).
- Ruolo di servizio: un ruolo di servizio è un [ruolo IAM](#) assunto da un servizio per eseguire operazioni per conto dell'utente. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione [Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS](#) nella Guida per l'utente di IAM.
- Ruolo collegato al servizio: un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un Servizio AWS. Il servizio può assumere il ruolo per eseguire un'azione per tuo conto. I ruoli collegati ai servizi sono visualizzati nell'account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non modificarle.
- Applicazioni in esecuzione su Amazon EC2: è possibile utilizzare un ruolo IAM per gestire credenziali temporanee per le applicazioni in esecuzione su un'istanza EC2 che eseguono richieste di AWS CLI o dell'API AWS. Ciò è preferibile all'archiviazione delle chiavi di accesso nell'istanza EC2. Per assegnare un ruolo AWS a un'istanza EC2, affinché sia disponibile per tutte le relative applicazioni, puoi creare un profilo dell'istanza collegato all'istanza. Un profilo dell'istanza contiene il ruolo e consente ai programmi in esecuzione sull'istanza EC2 di ottenere le credenziali temporanee. Per ulteriori informazioni, consulta [Utilizzo di un ruolo IAM per concedere autorizzazioni ad applicazioni in esecuzione su istanze di Amazon EC2](#) nella Guida per l'utente di IAM.

Per informazioni sull'utilizzo dei ruoli IAM, consulta [Quando creare un ruolo IAM \(invece di un utente\)](#) nella Guida per l'utente di IAM.

Gestione dell'accesso con policy

Per controllare l'accesso a AWS è possibile creare policy e collegarle a identità o risorse AWS. Una policy è un oggetto in AWS che, quando associato a un'identità o a una risorsa, ne definisce le autorizzazioni. AWS valuta queste policy quando un principale IAM (utente, utente root o sessione ruolo) effettua una richiesta. Le autorizzazioni nelle policy determinano l'approvazione o il rifiuto della richiesta. La maggior parte delle policy viene archiviata in AWS sotto forma di documenti JSON. Per ulteriori informazioni sulla struttura e sui contenuti dei documenti delle policy JSON, consulta [Panoramica delle policy JSON](#) nella Guida per l'utente di IAM.

Gli amministratori possono utilizzare le policy AWSJSON per specificare l'accesso ai diversi elementi. In altre parole, quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Per concedere agli utenti l'autorizzazione a eseguire azioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM. Successivamente l'amministratore può aggiungere le policy IAM ai ruoli e gli utenti possono assumere i ruoli.

Le policy IAM definiscono le autorizzazioni relative a un'operazione, a prescindere dal metodo utilizzato per eseguirla. Ad esempio, supponiamo di disporre di una policy che consente l'azione `iam:GetRole`. Un utente con tale policy può ottenere informazioni sul ruolo dalla AWS Management Console, la AWS CLI o l'API AWS.

Policy basate su identità

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le azioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta [Creazione di policy IAM](#) nella Guida per l'utente di IAM.

Le policy basate su identità possono essere ulteriormente classificate come policy inline o policy gestite. Le policy inline sono incorporate direttamente in un singolo utente, gruppo o ruolo. Le policy gestite sono policy autonome che possono essere collegate a più utenti, gruppi e ruoli in Account AWS. Le policy gestite includono le policy gestite da AWS e le policy gestite dal cliente. Per informazioni su come scegliere tra una policy gestita o una policy inline, consulta [Scelta fra policy gestite e policy inline](#) nella Guida per l'utente di IAM.

Policy basate su risorse

Le policy basate su risorse sono documenti di policy JSON che è possibile allegare a una risorsa. Gli esempi più comuni di policy basate su risorse sono le policy di attendibilità dei ruoli IAM e le policy dei bucket Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarle per controllare l'accesso a una risorsa specifica. Quando è allegata a una risorsa, una policy definisce le azioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario [specificare un principale](#) in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o Servizi AWS.

Le policy basate sulle risorse sono policy inline che si trovano in tale servizio. Non è possibile utilizzare le policy gestite da AWS da IAM in una policy basata su risorse.

Liste di controllo degli accessi (ACL)

Le liste di controllo degli accessi (ACL) controllano quali principali (membri, utenti o ruoli dell'account) hanno le autorizzazioni per accedere a una risorsa. Le ACL sono simili alle policy basate sulle risorse, sebbene non utilizzino il formato del documento di policy JSON.

Amazon S3, AWS WAF e Amazon VPC sono esempi di servizi che supportano le ACL. Per maggiori informazioni sulle ACL, consulta [Panoramica delle liste di controllo degli accessi \(ACL\)](#) nella Guida per gli sviluppatori di Amazon Simple Storage Service.

Altri tipi di policy

AWS supporta altri tipi di policy meno comuni. Questi tipi di policy possono impostare il numero massimo di autorizzazioni concesse dai tipi di policy più comuni.

- **Limiti delle autorizzazioni:** un limite delle autorizzazioni è una funzione avanzata nella quale si imposta il numero massimo di autorizzazioni che una policy basata su identità può concedere a un'entità IAM (utente o ruolo IAM). È possibile impostare un limite delle autorizzazioni per un'entità. Le autorizzazioni risultanti sono l'intersezione delle policy basate su identità dell'entità e i relativi limiti delle autorizzazioni. Le policy basate su risorse che specificano l'utente o il ruolo nel campo `Principal` sono condizionate dal limite delle autorizzazioni. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni sui limiti delle autorizzazioni, consulta [Limiti delle autorizzazioni per le entità IAM](#) nella Guida per l'utente di IAM.
- **Policy di controllo dei servizi (SCP):** le SCP sono policy JSON che specificano il numero massimo di autorizzazioni per un'organizzazione o unità organizzativa (OU) in AWS Organizations. AWS Organizations è un servizio per il raggruppamento e la gestione centralizzata degli Account

AWSmultipli di proprietà dell'azienda. Se abiliti tutte le funzionalità in un'organizzazione, puoi applicare le policy di controllo dei servizi (SCP) a uno o tutti i tuoi account. La SCP limita le autorizzazioni per le entità negli account membri, compreso ogni Utente root dell'account AWS. Per ulteriori informazioni su organizzazioni e policy SCP, consulta la pagina sulle [Policy di controllo dei servizi](#) nella Guida per l'utente di AWS Organizations.

- Policy di sessione: le policy di sessione sono policy avanzate che vengono trasmesse come parametro quando si crea in modo programmatico una sessione temporanea per un ruolo o un utente federato. Le autorizzazioni della sessione risultante sono l'intersezione delle policy basate su identità del ruolo o dell'utente e le policy di sessione. Le autorizzazioni possono anche provenire da una policy basata su risorse. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni, consulta [Policy di sessione](#) nella Guida per l'utente di IAM.

Più tipi di policy

Quando più tipi di policy si applicano a una richiesta, le autorizzazioni risultanti sono più complicate da comprendere. Per informazioni su come AWS determina se consentire una richiesta quando sono coinvolti più tipi di policy, consulta [Logica di valutazione delle policy](#) nella Guida per l'utente di IAM.

Funzionamento di Amazon ElastiCache con IAM

Prima di utilizzare IAM per gestire l'accesso a ElastiCache, scopri quali funzionalità di IAM sono disponibili per l'uso con ElastiCache.

Funzionalità IAM utilizzabili con Amazon ElastiCache

Funzionalità IAM	Supporto ElastiCache
Policy basate su identità	Sì
Policy basate su risorse	No
Azioni di policy	Sì
Risorse relative alle policy	Sì
Chiavi di condizione delle policy	Sì

Funzionalità IAM	Supporto ElastiCache
Liste di controllo degli accessi (ACL)	Sì
ABAC (tag nelle policy)	Sì
Credenziali temporanee	Sì
Autorizzazioni del principale	Sì
Ruoli di servizio	Sì
Ruoli collegati al servizio	Sì

Per ottenere un quadro generale del funzionamento di ElastiCache e altri servizi AWS con la maggior parte delle caratteristiche di IAM, consulta [Servizi AWS supportati da IAM](#) nella Guida per l'utente IAM.

Policy basate su identità per ElastiCache

Supporta le policy basate su identità	Sì
---------------------------------------	----

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le azioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta [Creazione di policy IAM](#) nella Guida per l'utente di IAM.

Con le policy basate su identità di IAM, è possibile specificare quali operazioni e risorse sono consentite o respinte, nonché le condizioni in base alle quali le operazioni sono consentite o respinte. Non è possibile specificare l'entità principale in una policy basata sull'identità perché si applica all'utente o al ruolo a cui è associato. Per informazioni su tutti gli elementi utilizzabili in una policy JSON, consulta [Guida di riferimento agli elementi delle policy JSON IAM](#) nella Guida per l'utente di IAM.

Esempi di policy basate su identità per ElastiCache

Per visualizzare esempi di policy basate su identità ElastiCache, consultare [Esempi di policy basate su identità per Amazon ElastiCache](#).

Policy basate su risorse all'interno di ElastiCache

Supporta le policy basate su risorse

No

Le policy basate su risorse sono documenti di policy JSON che è possibile allegare a una risorsa. Gli esempi più comuni di policy basate su risorse sono le policy di attendibilità dei ruoli IAM e le policy dei bucket Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarle per controllare l'accesso a una risorsa specifica. Quando è allegata a una risorsa, una policy definisce le azioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario [specificare un principale](#) in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o Servizi AWS.

Per consentire l'accesso multi-account, puoi specificare un intero account o entità IAM in un altro account come principale in una policy basata sulle risorse. L'aggiunta di un principale multi-account a una policy basata sulle risorse rappresenta solo una parte della relazione di trust. Quando l'entità principale e la risorsa si trovano in diversi Account AWS, un amministratore IAM nell'account attendibile deve concedere all'entità principale (utente o ruolo) anche l'autorizzazione per accedere alla risorsa. L'autorizzazione viene concessa collegando all'entità una policy basata sull'identità. Tuttavia, se una policy basata su risorse concede l'accesso a un principale nello stesso account, non sono richieste ulteriori policy basate su identità. Per ulteriori informazioni, consulta [Differenza tra i ruoli IAM e le policy basate su risorse](#) nella Guida per l'utente di IAM.

Operazioni delle policy per ElastiCache

Supporta le azioni di policy

Sì

Gli amministratori possono utilizzare le policy JSON AWS per specificare gli accessi ai diversi elementi. Cioè, quale principale può eseguire azioni su quali risorse, e in quali condizioni.

L'elemento `Action` di una policy JSON descrive le operazioni che è possibile utilizzare per consentire o negare l'accesso a un criterio. Le azioni di policy hanno spesso lo stesso nome dell'operazione API AWS. Ci sono alcune eccezioni, ad esempio le azioni di sola autorizzazione che non hanno un'operazione API corrispondente. Esistono anche alcune operazioni che richiedono più operazioni in una policy. Queste operazioni aggiuntive sono denominate operazioni dipendenti.

Includi le operazioni in una policy per concedere le autorizzazioni a eseguire l'operazione associata.

Per visualizzare un elenco di azioni ElastiCache, consulta [Operazioni definite da Amazon ElastiCache](#) nella Guida di riferimento per l'autorizzazione del servizio.

Le operazioni delle policy in ElastiCache utilizzano il seguente prefisso prima dell'operazione:

```
elasticache
```

Per specificare più operazioni in una sola istruzione, occorre separarle con la virgola.

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

È possibile specificare più operazioni tramite caratteri jolly (*). Ad esempio, per specificare tutte le operazioni che iniziano con la parola Describe, includi la seguente operazione:

```
"Action": "elasticache:Describe*"
```

Per visualizzare esempi di policy basate su identità ElastiCache, consultare [Esempi di policy basate su identità per Amazon ElastiCache](#).

Risorse della policy per ElastiCache

Supporta le risorse di policy	Sì
-------------------------------	----

Gli amministratori possono utilizzare le policy JSON AWS per specificare gli accessi ai diversi elementi. Cioè, quale principale può eseguire operazioni su quali risorse, e in quali condizioni.

L'elemento JSON Resource della policy specifica l'oggetto o gli oggetti ai quali si applica l'azione. Le istruzioni devono includere un elemento Resource o un elemento NotResource. Come best practice, specifica una risorsa utilizzando il suo [nome della risorsa Amazon \(ARN\)](#). Puoi eseguire questa operazione per azioni che supportano un tipo di risorsa specifico, note come autorizzazioni a livello di risorsa.

Per le azioni che non supportano le autorizzazioni a livello di risorsa, ad esempio le operazioni di elenco, utilizza un carattere jolly (*) per indicare che l'istruzione si applica a tutte le risorse.

```
"Resource": "*"

```

Per visualizzare un elenco di tipi di risorse ElastiCache e i relativi ARN, consulta [Tipi di risorsa definiti da Amazon ElastiCache](#) nella Guida di riferimento per l'autorizzazione del servizio. Per informazioni sulle operazioni con cui è possibile specificare l'ARN di ogni risorsa, consulta [Operazioni definite da Amazon ElastiCache](#).

Per visualizzare esempi di policy basate su identità ElastiCache, consultare [Esempi di policy basate su identità per Amazon ElastiCache](#).

Chiavi di condizione delle policy per ElastiCache

Supporta le chiavi di condizione delle policy specifiche del servizio	Si
---	----

Gli amministratori possono utilizzare le policy JSON AWS per specificare gli accessi ai diversi elementi. Cioè, quale principale può eseguire operazioni su quali risorse, e in quali condizioni.

L'elemento `Condition` (o blocco `Condition`) consente di specificare le condizioni in cui un'istruzione è in vigore. L'elemento `Condition` è facoltativo. Puoi compilare espressioni condizionali che utilizzano [operatori di condizione](#), ad esempio uguale a o minore di, per soddisfare la condizione nella policy con i valori nella richiesta.

Se specifichi più elementi `Condition` in un'istruzione o più chiavi in un singolo elemento `Condition`, questi vengono valutati da AWS utilizzando un'operazione AND logica. Se specifichi più valori per una singola chiave di condizione, AWS valuta la condizione utilizzando un'operazione OR logica. Tutte le condizioni devono essere soddisfatte prima che le autorizzazioni dell'istruzione vengano concesse.

Puoi anche utilizzare variabili segnaposto quando specifichi le condizioni. Ad esempio, puoi autorizzare un utente IAM ad accedere a una risorsa solo se è stata taggata con il relativo nome utente IAM. Per ulteriori informazioni, consulta [Elementi delle policy IAM: variabili e tag](#) nella Guida per l'utente di IAM.

AWS supporta chiavi di condizione globali e chiavi di condizione specifiche per il servizio. Per visualizzare tutte le chiavi di condizione globali di AWS, consulta [Chiavi di contesto delle condizioni globali di AWS](#) nella Guida per l'utente di IAM.

Per un elenco delle chiavi di condizione di ElastiCache, consulta [Chiavi di condizione per Amazon ElastiCache](#) nella Guida di riferimento per l'autorizzazione del servizio. Per informazioni su operazioni e risorse con cui è possibile utilizzare una chiave di condizione, consulta [Operazioni definite da Amazon ElastiCache](#).

Per visualizzare esempi di policy basate su identità ElastiCache, consultare [Esempi di policy basate su identità per Amazon ElastiCache](#).

Liste di controllo degli accessi (ACL) in ElastiCache

Supporta le ACL

Sì

Le liste di controllo degli accessi (ACL) controllano quali principali (membri, utenti o ruoli dell'account) hanno le autorizzazioni ad accedere a una risorsa. Le ACL sono simili alle policy basate su risorse, sebbene non utilizzino il formato del documento di policy JSON.

Controllo degli accessi basato su attributi (ABAC) con ElastiCache

Supporta ABAC (tag nelle policy)

Sì

Il controllo dell'accesso basato su attributi (ABAC) è una strategia di autorizzazione che definisce le autorizzazioni in base agli attributi. In AWS, tali attributi sono denominati tag. È possibile collegare dei tag alle entità IAM (utenti o ruoli) e a numerose risorse AWS. L'assegnazione di tag alle entità e alle risorse è il primo passaggio di ABAC. In seguito, vengono progettate policy ABAC per consentire operazioni quando il tag dell'entità principale corrisponde al tag sulla risorsa a cui si sta provando ad accedere.

La strategia ABAC è utile in ambienti soggetti a una rapida crescita e aiuta in situazioni in cui la gestione delle policy diventa impegnativa.

Per controllare l'accesso basato su tag, fornisci informazioni sui tag nell'[elemento condizione](#) di una policy utilizzando le chiavi di condizione `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`.

Se un servizio supporta tutte e tre le chiavi di condizione per ogni tipo di risorsa, il valore per il servizio è Yes (Sì). Se un servizio supporta tutte e tre le chiavi di condizione solo per alcuni tipi di risorsa, allora il valore sarà Parziale.

Per ulteriori informazioni su ABAC, consulta [Che cos'è ABAC?](#) nella Guida per l'utente di IAM. Per visualizzare un tutorial con i passaggi per l'impostazione di ABAC, consulta [Utilizzo del controllo degli accessi basato su attributi \(ABAC\)](#) nella Guida per l'utente di IAM.

Utilizzo di credenziali temporanee con Elasticache

Supporta le credenziali temporanee	Sì
------------------------------------	----

Alcuni Servizi AWS non funzionano quando si accede utilizzando credenziali temporanee. Per ulteriori informazioni, inclusi i Servizi AWS che funzionano con le credenziali temporanee, consulta [Servizi AWS supportati da IAM](#) nella Guida per l'utente IAM.

Le credenziali temporanee sono utilizzate se si accede alla AWS Management Console utilizzando qualsiasi metodo che non sia la combinazione di nome utente e password. Ad esempio, quando accedi ad AWS utilizzando il collegamento Single Sign-On (SSO) della tua azienda, tale processo crea in automatico credenziali temporanee. Le credenziali temporanee vengono create in automatico anche quando accedi alla console come utente e poi cambi ruolo. Per ulteriori informazioni sullo scambio dei ruoli, consulta [Cambio di un ruolo \(console\)](#) nella Guida per l'utente di IAM.

È possibile creare manualmente credenziali temporanee utilizzando la AWS CLI o l'API AWS. È quindi possibile utilizzare tali credenziali temporanee per accedere ad AWS. AWS consiglia di generare le credenziali temporanee dinamicamente anziché utilizzare chiavi di accesso a lungo termine. Per ulteriori informazioni, consulta [Credenziali di sicurezza provvisorie in IAM](#).

Autorizzazioni del principale tra servizi per Elasticache

Supporta sessioni di accesso diretto (FAS)	Sì
--	----

Quando si utilizza un utente o un ruolo IAM per eseguire operazioni in AWS, si viene considerati un principale. Quando si utilizzano alcuni servizi, è possibile eseguire un'azione che attiva un'altra azione in un servizio diverso. FAS utilizza le autorizzazioni del principale che effettua la chiamata a un Servizio AWS, combinate con il Servizio AWS richiedente, per effettuare richieste a servizi a valle. Le richieste FAS vengono effettuate solo quando un servizio riceve una richiesta che necessita di interazioni con altri Servizi AWS o risorse per essere portata a termine. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le operazioni. Per i dettagli delle policy relative alle richieste FAS, consulta la pagina [Forward access sessions](#).

Ruoli di servizio per ElastiCache

Supporta i ruoli di servizio	Sì
------------------------------	----

Un ruolo di servizio è un [ruolo IAM](#) che un servizio assume per eseguire operazioni per tuo conto. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione [Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS](#) nella Guida per l'utente di IAM.

Warning

La modifica delle autorizzazioni per un ruolo di servizio potrebbe compromettere la funzionalità di Elasticache. Modificare i ruoli del servizio solo quando Elasticache fornisce le indicazioni per farlo.

Ruoli collegati ai servizi per ElastiCache

Supporta i ruoli collegati ai servizi	Sì
---------------------------------------	----

Un ruolo collegato ai servizi è un tipo di ruolo di servizio che è collegato a un Servizio AWS. Il servizio può assumere il ruolo per eseguire un'operazione per tuo conto. I ruoli collegati ai servizi sono visualizzati nell'account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non modificarle.

Per ulteriori informazioni su come creare e gestire i ruoli collegati ai servizi, consulta [Servizi AWS supportati da IAM](#). Trova un servizio nella tabella che include un Yes nella colonna Service-linked role (Ruolo collegato ai servizi). Scegli il collegamento Sì per visualizzare la documentazione relativa al ruolo collegato ai servizi per tale servizio.

Esempi di policy basate su identità per Amazon ElastiCache

Per impostazione predefinita, gli utenti e i ruoli non dispongono dell'autorizzazione per creare o modificare le risorse ElastiCache. Inoltre, non sono in grado di eseguire attività utilizzando la AWS Management Console, l'AWS Command Line Interface (AWS CLI) o l'API AWS. Per concedere agli

utenti l'autorizzazione per eseguire operazioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM. L'amministratore può quindi aggiungere le policy IAM ai ruoli e gli utenti possono assumere i ruoli.

Per informazioni su come creare una policy basata su identità IAM utilizzando questi documenti di policy JSON di esempio, consulta [Creazione di policy IAM](#) nella Guida per l'utente di IAM.

Per informazioni dettagliate sulle operazioni e sui tipi di risorse definiti da ElastiCache, incluso il formato degli ARN per ogni tipo di risorsa, consultare [Operazioni, risorse e chiavi di condizione per Amazon ElastiCache](#) nella Guida di riferimento per l'autorizzazione del servizio.

Argomenti

- [Best practice per le policy](#)
- [Utilizzo della console ElastiCache](#)
- [Consentire agli utenti di visualizzare le loro autorizzazioni](#)

Best practice per le policy

Le policy basate su identità determinano se qualcuno può creare, accedere o eliminare risorse ElastiCache nel tuo account. Queste operazioni possono comportare costi aggiuntivi per l'Account AWS. Quando crei o modifichi policy basate su identità, segui queste linee guida e raccomandazioni:

- Nozioni di base sulle policy gestite da AWS e passaggio alle autorizzazioni con privilegio minimo: per le informazioni di base su come concedere autorizzazioni a utenti e carichi di lavoro, utilizza le policy gestite da AWS che concedono le autorizzazioni per molti casi d'uso comuni. Sono disponibili nel tuo Account AWS. Ti consigliamo pertanto di ridurre ulteriormente le autorizzazioni definendo policy gestite dal cliente di AWS specifiche per i tuoi casi d'uso. Per ulteriori informazioni, consulta [Policy gestite da AWS](#) o [Policy gestite da AWS per le funzioni dei processi](#) nella Guida per l'utente IAM.
- Applica le autorizzazioni con privilegi minimi: quando imposti le autorizzazioni con le policy IAM, concedi solo le autorizzazioni richieste per eseguire un'attività. Puoi farlo definendo le azioni che possono essere intraprese su risorse specifiche in condizioni specifiche, note anche come autorizzazioni con privilegi minimi. Per ulteriori informazioni sull'utilizzo di IAM per applicare le autorizzazioni, consulta [Policy e autorizzazioni in IAM](#) nella Guida per l'utente di IAM.
- Condizioni d'uso nelle policy IAM per limitare ulteriormente l'accesso: per limitare l'accesso a operazioni e risorse puoi aggiungere una condizione alle tue policy. Ad esempio, è possibile scrivere una condizione di policy per specificare che tutte le richieste devono essere inviate

utilizzando SSL. Puoi inoltre utilizzare le condizioni per concedere l'accesso alle operazioni di servizio, ma solo se vengono utilizzate tramite uno specifico Servizio AWS, ad esempio AWS CloudFormation. Per ulteriori informazioni, consulta la sezione [Elementi delle policy JSON di IAM: condizione](#) nella Guida per l'utente di IAM.

- Utilizzo di IAM Access Analyzer per convalidare le policy IAM e garantire autorizzazioni sicure e funzionali: IAM Access Analyzer convalida le policy nuove ed esistenti in modo che aderiscano alla sintassi della policy IAM (JSON) e alle best practice di IAM. IAM Access Analyzer offre oltre 100 controlli delle policy e consigli utili per creare policy sicure e funzionali. Per ulteriori informazioni, consulta [Convalida delle policy per IAM Access Analyzer](#) nella Guida per l'utente di IAM.
- Richiesta dell'autenticazione a più fattori (MFA): se hai uno scenario che richiede utenti IAM o utenti root nel tuo Account AWS, attiva MFA per una maggiore sicurezza. Per richiedere la MFA quando vengono chiamate le operazioni API, aggiungi le condizioni MFA alle policy. Per ulteriori informazioni, consulta [Configurazione dell'accesso alle API protetto con MFA](#) nella Guida per l'utente di IAM.

Per maggiori informazioni sulle best practice in IAM, consulta [Best practice di sicurezza in IAM](#) nella Guida per l'utente di IAM.

Utilizzo della console ElastiCache

Per accedere alla console Amazon ElastiCache, è necessario disporre di un set di autorizzazioni minimo. Queste autorizzazioni devono consentire di elencare e visualizzare i dettagli relativi alle risorse ElastiCache nel tuo Account AWS. Se crei una policy basata sull'identità più restrittiva rispetto alle autorizzazioni minime richieste, la console non funzionerà nel modo previsto per le entità (utenti o ruoli) associate a tale policy.

Non è necessario concedere le autorizzazioni minime della console agli utenti che effettuano chiamate solo alla AWS CLI o all'API AWS. Al contrario, concedi l'accesso solo alle operazioni che corrispondono all'operazione API che stanno cercando di eseguire.

Per garantire che gli utenti e i ruoli possano continuare a utilizzare la console ElastiCache, collega anche la policy gestita ConsoleAccess o ReadOnly AWS di ElastiCache alle entità. Per ulteriori informazioni, consulta [Aggiunta di autorizzazioni a un utente](#) nella Guida per l'utente IAM.

Consentire agli utenti di visualizzare le loro autorizzazioni

Questo esempio mostra in che modo è possibile creare una policy che consente agli utenti IAM di visualizzare le policy inline e gestite che sono allegate alla relativa identità utente. La policy include le

autorizzazioni per completare questa azione sulla console o a livello di programmazione utilizzando la AWS CLI o l'API AWS.

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

Risoluzione dei problemi relativi all'identità e all'accesso di Amazon ElastiCache

Utilizza le informazioni seguenti per diagnosticare e risolvere i problemi comuni che possono verificarsi durante l'utilizzo di ElastiCache e di IAM.

Argomenti

- [Non sono autorizzato/a a eseguire un'operazione in ElastiCache](#)
- [Non sono autorizzato a eseguire iam:PassRole](#)
- [Voglio consentire alle persone esterne al mio account AWS di accedere alle mie risorse ElastiCache](#)

Non sono autorizzato/a a eseguire un'operazione in ElastiCache

Se la AWS Management Console indica che non sei autorizzato a eseguire un'operazione, devi contattare l'amministratore per ricevere assistenza. L'amministratore è la persona che ti ha fornito il nome utente e la password.

Il seguente esempio di errore si verifica quando l'utente `mateojackson` prova a utilizzare la console per visualizzare i dettagli relativi a una risorsa `my-example-widget` fittizia, ma non dispone di autorizzazioni `elasticache:GetWidget` fittizie.

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

In questo caso, Mateo richiede al suo amministratore di aggiornare le sue policy per poter accedere alla risorsa `my-example-widget` utilizzando l'operazione `elasticache:GetWidget`.

Non sono autorizzato a eseguire iam:PassRole

Se ricevi un errore che indica che non sei autorizzato a eseguire l'operazione `iam:PassRole`, devi aggiornare le policy per poter passare un ruolo a ElastiCache.

Alcuni Servizi AWS consentono di passare un ruolo esistente a tale servizio, invece di creare un nuovo ruolo di servizio o un ruolo collegato ai servizi. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per passare il ruolo al servizio.

L'errore di esempio riportato di seguito si verifica quando un utente IAM denominato `marymajor` tenta di utilizzare la console per eseguire un'operazione in ElastiCache. Tuttavia, l'operazione

richiede che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per passare il ruolo al servizio.

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

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Per ulteriore assistenza con l'accesso, contatta l'amministratore AWS. L'amministratore è colui che ti ha fornito le credenziali di accesso.

Voglio consentire alle persone esterne al mio account AWS di accedere alle mie risorse ElastiCache

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle tue risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo. Per servizi che supportano policy basate su risorse o liste di controllo accessi (ACL), utilizza tali policy per concedere alle persone l'accesso alle tue risorse.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- Per sapere se ElastiCache supporta queste funzionalità, consultare [Funzionamento di Amazon ElastiCache con IAM](#).
- Per informazioni su come garantire l'accesso alle risorse negli Account AWS che possiedi, consulta [Fornire l'accesso a un utente IAM in un altro Account AWS in tuo possesso](#) nella Guida per l'utente di IAM.
- Per informazioni su come fornire l'accesso alle risorse ad Account AWS di terze parti, consulta [Fornire l'accesso agli Account AWS di proprietà di terze parti](#) nella Guida per l'utente di IAM.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consulta [Fornire l'accesso a utenti autenticati esternamente \(Federazione delle identità\)](#) nella Guida per l'utente di IAM.
- Per informazioni sulle differenze tra l'utilizzo di ruoli e policy basate su risorse per l'accesso multi-account, consultare [Differenza tra i ruoli IAM e le policy basate su risorse](#) nella Guida per l'utente di IAM.

Controllo accessi

Per autenticare le richieste, è necessario disporre di credenziali valide, ma a meno che non si disponga delle autorizzazioni non è possibile creare o accedere a risorse di ElastiCache. Ad esempio, occorre disporre delle autorizzazioni per creare un cluster ElastiCache.

Nelle sezioni seguenti viene descritto come gestire le autorizzazioni per ElastiCache. Consigliamo di leggere prima la panoramica.

- [Panoramica della gestione delle autorizzazioni di accesso alle risorse di ElastiCache](#)
- [Utilizzo di policy basate su identità \(policy IAM\) per Amazon ElastiCache](#)

Panoramica della gestione delle autorizzazioni di accesso alle risorse di ElastiCache

Ogni risorsa AWS è di proprietà di un account AWS e le autorizzazioni necessarie per creare o accedere a una risorsa sono regolate dalle policy di autorizzazione. Un amministratore dell'account è in grado di collegare le policy relative alle autorizzazioni alle identità IAM (ovvero utenti, gruppi e ruoli). Inoltre, Amazon ElastiCache supporta anche il collegamento di policy di autorizzazione alle risorse.

Note

Un amministratore account (o un utente amministratore) è un utente con privilegi di amministratore. Per ulteriori informazioni, consulta [Best practice IAM](#) nella Guida per l'utente di IAM.

Per fornire l'accesso, aggiungi autorizzazioni ai tuoi utenti, gruppi o ruoli:

- Utenti e gruppi in AWS IAM Identity Center:

Crea un set di autorizzazioni. Segui le istruzioni riportate nella pagina [Create a permission set](#) (Creazione di un set di autorizzazioni) nella Guida per l'utente di AWS IAM Identity Center.

- Utenti gestiti in IAM tramite un provider di identità:

Crea un ruolo per la federazione delle identità. Segui le istruzioni riportate nella pagina [Creating a role for a third-party identity provider \(federation\)](#) (Creazione di un ruolo per un provider di identità di terze parti [federazione]) nella Guida per l'utente di IAM.

- Utenti IAM:

- Crea un ruolo che l'utente possa assumere. Per istruzioni, consulta la pagina [Creating a role for an IAM user](#) (Creazione di un ruolo per un utente IAM) nella Guida per l'utente di IAM.
- (Non consigliato) Collega una policy direttamente a un utente o aggiungi un utente a un gruppo di utenti. Segui le istruzioni riportate nella pagina [Aggiunta di autorizzazioni a un utente \(console\)](#) nella Guida per l'utente di IAM.

Argomenti

- [Risorse e operazioni di Amazon ElastiCache](#)

- [Informazioni sulla proprietà delle risorse](#)
- [Gestione dell'accesso alle risorse](#)
- [Policy gestite da AWS per Amazon ElastiCache](#)
- [Utilizzo di policy basate su identità \(policy IAM\) per Amazon ElastiCache](#)
- [Autorizzazioni a livello di risorsa](#)
- [Utilizzo delle chiavi di condizione](#)
- [Utilizzo di ruoli collegati ai servizi per Amazon ElastiCache](#)
- [Autorizzazioni API di ElastiCache: riferimento a operazioni, risorse e condizioni](#)

Risorse e operazioni di Amazon ElastiCache

Per visualizzare un elenco di tipi di risorse ElastiCache e i relativi ARN, consulta [Tipi di risorsa definiti da Amazon ElastiCache](#) nella Guida di riferimento per l'autorizzazione del servizio. Per informazioni sulle operazioni con cui è possibile specificare l'ARN di ogni risorsa, consulta [Operazioni definite da Amazon ElastiCache](#).

Informazioni sulla proprietà delle risorse

Il proprietario di una risorsa è l'account AWS che crea la risorsa. Ovvero, il proprietario di una risorsa è l'account AWS dell'entità principale che esegue l'autenticazione della richiesta che ha creato la risorsa. Un'entità principale può essere l'account root, un utente IAM o un ruolo IAM. Negli esempi seguenti viene illustrato il funzionamento:

- Supponi di utilizzare le credenziali dell'account root del tuo account AWS per creare un cluster di cache. In questo caso, il tuo account AWS è il proprietario della risorsa. In ElastiCache, la risorsa è il cluster di cache
- Supponi di creare un utente IAM nel tuo account AWS e di concedere a tale utente le autorizzazioni necessarie per creare un cluster di cache. In questo caso, l'utente può creare un cluster di cache. Tieni presente, tuttavia, che l'account AWS a cui appartiene l'utente è il proprietario della risorsa cluster di cache.
- Supponi di creare un ruolo IAM nel tuo account AWS con le autorizzazioni necessarie per creare un cluster di cache. In questo caso, chiunque sia in grado di assumere il ruolo può creare un cluster di cache. L'account AWS a cui appartiene il ruolo è il proprietario della risorsa cluster di cache.

Gestione dell'accesso alle risorse

La policy delle autorizzazioni descrive chi ha accesso a cosa. Nella sezione seguente vengono descritte le opzioni disponibili per la creazione di policy relative alle autorizzazioni.

Note

In questa sezione viene descritto l'utilizzo di IAM nel contesto di Amazon ElastiCache. Non vengono fornite informazioni dettagliate sul servizio IAM. Per la documentazione di IAM completa, consulta [Che cos'è IAM?](#) nella Guida per l'utente di IAM. Per informazioni sulla sintassi delle policy IAM e le rispettive descrizioni, consulta [Riferimento alle policy IAM di AWS](#) nella Guida per l'utente di IAM.

Le policy collegate a un'identità IAM vengono definite policy basate su identità (policy IAM). Le policy collegate a una risorsa vengono definite policy basate sulle risorse.

Argomenti

- [Policy basate su identità \(policy IAM\)](#)
- [Specifica degli elementi delle policy: operazioni, effetti, risorse ed entità](#)
- [Specifica delle condizioni in una policy](#)

Policy basate su identità (policy IAM)

Puoi collegare le policy alle identità IAM. Ad esempio, puoi eseguire le operazioni seguenti:

- Collegare una policy di autorizzazione a un utente o a un gruppo nell'account – Per assegnare le autorizzazioni un amministratore di account può utilizzare una policy di autorizzazione associata a un utente specifico. In questo caso, le autorizzazioni permettono a tale utente di creare una risorsa ElastiCache, come un cluster di cache, un gruppo di parametri o un gruppo di sicurezza.
- Collega una policy di autorizzazione a un ruolo (assegnazione di autorizzazioni tra account): per concedere autorizzazioni multi-account, è possibile collegare una policy di autorizzazione basata su identità a un ruolo IAM. Ad esempio, l'amministratore dell'account A può creare un ruolo per concedere autorizzazioni multi-account a un altro account AWS (ad esempio l'account B) oppure a un servizio AWS nel modo seguente:
 1. L'amministratore dell'account A crea un ruolo IAM e attribuisce una policy di autorizzazione al ruolo che concede le autorizzazioni sulle risorse per l'account A.

2. L'amministratore dell'account A attribuisce una policy di attendibilità al ruolo, identificando l'account B come il principale per tale ruolo.
3. L'amministratore dell'account B può quindi delegare le autorizzazioni ad assumere tale ruolo a qualsiasi utente dell'account B. In questo modo, gli utenti nell'account B possono creare o accedere alle risorse nell'account A. In alcuni casi potrebbe essere opportuno concedere le autorizzazioni per assumere tale ruolo a un servizio AWS. Per supportare tale approccio, l'entità principale nella policy di trust può anche essere un'entità principale di un servizio AWS.

Per ulteriori informazioni sull'uso di IAM per delegare le autorizzazioni, consulta [Access Management](#) nella IAM User Guide (Guida per l'utente di IAM).

Di seguito è riportato un esempio di policy che consente a un utente di eseguire l'operazione `DescribeCacheClusters` per il tuo account AWS. ElastiCache supporta anche l'identificazione di risorse specifiche utilizzando gli ARN delle risorse per le operazioni API. (questo approccio è anche noto come autorizzazioni a livello di risorsa).

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "DescribeCacheClusters",
    "Effect": "Allow",
    "Action": [
      "elasticache:DescribeCacheClusters"],
    "Resource": resource-arn
  ]
}
```

Per ulteriori informazioni sull'uso di policy basate su identità con ElastiCache, consulta [Utilizzo di policy basate su identità \(policy IAM\) per Amazon ElastiCache](#). Per ulteriori informazioni su utenti, gruppi, ruoli e autorizzazioni, consulta [Identità \(utenti, gruppi e ruoli\)](#) nella Guida per l'utente di IAM.

Specifica degli elementi delle policy: operazioni, effetti, risorse ed entità

Per ogni risorsa di Amazon ElastiCache (consulta [Risorse e operazioni di Amazon ElastiCache](#)), il servizio definisce un insieme di operazioni API (consulta [Operazioni](#)). Per concedere le autorizzazioni per queste operazioni API, ElastiCache definisce un set di operazioni che possono essere specificate in una policy. Ad esempio, per la risorsa cluster di ElastiCache, vengono definite le operazioni

`CreateCacheCluster`, `DeleteCacheCluster`, e `DescribeCacheCluster`. L'esecuzione di un'operazione API può richiedere le autorizzazioni per più di un'operazione.

Di seguito sono elencati gli elementi di base di una policy:

- **Risorsa:** in una policy si utilizza il nome della risorsa Amazon (ARN) per identificare la risorsa a cui si applica la policy stessa. Per ulteriori informazioni, consulta [Risorse e operazioni di Amazon ElastiCache](#).
- **Operazione:** utilizzi le parole chiave per identificare le operazioni sulla risorsa da permettere o rifiutare. Ad esempio, a seconda dell'Effect specificato, l'autorizzazione `elasticache:CreateCacheCluster` permette o nega all'utente le autorizzazioni per l'esecuzione dell'operazione di Amazon ElastiCache `CreateCacheCluster`.
- **Effetto:** l'effetto prodotto quando l'utente richiede l'operazione specifica, ovvero un'autorizzazione o un rifiuto. `Use` non concedi esplicitamente (consenti) l'accesso a una risorsa, l'accesso viene implicitamente rifiutato. È anche possibile negare esplicitamente l'accesso a una risorsa. Ad esempio, è possibile eseguire questa operazione per accertarsi che un utente non sia in grado di accedere a una risorsa, anche se l'accesso viene concesso da un'altra policy.
- **Principale:** nelle policy basate su identità (policy IAM), l'utente a cui la policy è collegata è il principale implicito. Per policy basate su risorse, specifichi l'utente, l'account, il servizio o un'altra entità che desideri riceva le autorizzazioni (si applica solo alle policy basate su risorse).

Per ulteriori informazioni sulla sintassi e le descrizioni delle policy IAM, consulta [AWSRiferimento alle policy IAM](#) nella Guida per l'utente di IAM.

Per una tabella che include tutte le operazioni API di Amazon ElastiCache, consulta [Autorizzazioni API di ElastiCache: riferimento a operazioni, risorse e condizioni](#).

Specifiche delle condizioni in una policy

Quando si concedono le autorizzazioni, è possibile utilizzare il linguaggio della policy IAM per specificare le condizioni in base a cui la policy deve essere applicata. Ad esempio, potresti decidere che una policy venga applicata solo dopo una data specifica. Per ulteriori informazioni su come specificare le condizioni in un linguaggio di policy, consulta la sezione [Condizione](#) nella Guida per l'utente di IAM.

Per esprimere le condizioni è necessario utilizzare chiavi di condizione predefinite. Per usare le chiavi di condizione specifiche di ElastiCache, consulta [Utilizzo delle chiavi di condizione](#). Tuttavia, ci sono

chiavi di condizione ampie AWS disponibili che puoi utilizzare secondo necessità. Per un elenco completo di chiavi AWS, consulta [Chiavi disponibili per le condizioni](#) nella Guida per l'utente di IAM.

Policy gestite da AWS per Amazon ElastiCache

Una policy gestita da AWS è una policy autonoma creata e amministrata da AWS. Le policy gestite da AWS sono progettate per fornire autorizzazioni per molti casi d'uso comuni in modo da poter iniziare ad assegnare autorizzazioni a utenti, gruppi e ruoli.

Ricorda che le policy gestite da AWS potrebbero non concedere autorizzazioni con privilegi minimi per i tuoi casi d'uso specifici perché possono essere utilizzate da tutti i clienti AWS. Consigliamo pertanto di ridurre ulteriormente le autorizzazioni definendo [policy gestite dal cliente](#) specifiche per i tuoi casi d'uso.

Non è possibile modificare le autorizzazioni definite nelle policy gestite da AWS. Se AWS aggiorna le autorizzazioni definite in una policy gestita da AWS, l'aggiornamento riguarda tutte le identità principali (utenti, gruppi e ruoli) a cui è collegata la policy. È molto probabile che AWS aggiorni una policy gestita da AWS quando viene lanciato un nuovo Servizio AWS o nuove operazioni API diventano disponibili per i servizi esistenti.

Per ulteriori informazioni, consultare [Policy gestite da AWS](#) nella Guida per l'utente di IAM.

Policy gestita da AWS: ElastiCacheServiceRolePolicy

Non è possibile collegare la policy ElastiCacheServiceRolePolicy alle entità IAM. Questa policy è associata a un ruolo collegato ai servizi che consente a ElastiCache di eseguire operazioni per tuo conto.

Questa policy consente a ElastiCache di gestire le risorse AWS per tuo conto, se necessario per la gestione della cache:

- `ec2`: gestisci le risorse di rete EC2 da collegare ai nodi di cache, inclusi endpoint VPC (per cache serverless), interfacce di rete elastica (ENI) (per cluster progettati autonomamente) e gruppi di sicurezza.
- `cloudwatch`: genera i dati delle metriche del servizio in CloudWatch.
- `outposts`: consenti la creazione di nodi di cache su AWS Outposts.

La policy [ElastiCacheServiceRolePolicy](#) è disponibile nella console IAM e in [ElastiCacheServiceRolePolicy](#) nella Guida di riferimento delle policy gestite da AWS.

Policy gestita da AWS: AmazonElastiCacheFullAccess

È possibile allegare la policy AmazonElastiCacheFullAccess alle identità IAM.

Questa policy consente ai principali l'accesso completo a ElastiCache utilizzando la Console di gestione AWS:

- `elasticache`: accedi a tutte le API.
- `iam`: crea il ruolo collegato ai servizi necessario per il funzionamento dei servizi.
- `ec2`: descrivi le risorse EC2 dipendenti necessarie per la creazione della cache (VPC, sottorete, gruppo di sicurezza) e consenti la creazione di endpoint VPC (per cache serverless).
- `kms`: consenti l'utilizzo di CMK gestite dal cliente per la crittografia dei dati a riposo.
- `cloudwatch`: consenti l'accesso alle metriche per visualizzare le metriche di ElastiCache nella console.
- `application-autoscaling`: consenti l'accesso per descrivere le policy di dimensionamento automatico per le cache.
- `logs`: utilizzato per popolare i flussi di log per la funzionalità di distribuzione dei log nella console.
- `firehose`: utilizzato per popolare i flussi di distribuzione per la funzionalità di distribuzione dei log nella console.
- `s3`: utilizzato per popolare i bucket S3 per la funzionalità di ripristino degli snapshot nella console.
- `outposts`: utilizzato per popolare AWS Outposts per la creazione di cache nella console.
- `sns`: utilizzato per popolare gli argomenti SNS per la funzionalità di notifica nella console.

La policy [AmazonElastiCacheFullAccess](#) è disponibile nella console IAM e in [AmazonElastiCacheFullAccess](#) nella Guida di riferimento delle policy gestite da AWS.

Policy gestita da AWS: AmazonElastiCacheReadOnlyAccess

È possibile allegare la policy AmazonElastiCacheReadOnlyAccess alle identità IAM.

Questa policy consente ai principali l'accesso in sola lettura a ElastiCache utilizzando la Console di gestione AWS:

- `elasticache`: accedi alle API `Describe` di sola lettura.

La policy [AmazonElastiCacheReadOnlyAccess](#) è disponibile nella console IAM e in [AmazonElastiCacheReadOnlyAccess](#) nella Guida di riferimento delle policy gestite da AWS.

Aggiornamenti di ElastiCache alle policy gestite da AWS

Visualizza i dettagli sugli aggiornamenti alle policy gestite da AWS per ElastiCache a partire da quando il servizio ha iniziato a monitorare le modifiche. Per gli avvisi automatici sulle modifiche apportate a questa pagina, sottoscrivi il feed RSS nella pagina della cronologia dei documenti di ElastiCache.

Modifica	Descrizione	Data
AmazonElastiCacheFullAccess - Aggiornamento di una policy esistente	ElastiCache ha aggiunto nuove autorizzazioni per consentire la gestione delle cache serverless e l'utilizzo di tutte le funzionalità del servizio tramite la console.	27 novembre 2023
ElastiCacheServiceRolePolicy - Aggiornamento di una policy esistente	ElastiCache ha aggiunto nuove autorizzazioni per consentire la gestione degli endpoint VPC per le risorse di cache serverless.	27 novembre 2023
ElastiCache ha iniziato a monitorare le modifiche	ElastiCache ha iniziato a monitorare le modifiche per le policy gestite da AWS.	7 febbraio 2020

Utilizzo di policy basate su identità (policy IAM) per Amazon ElastiCache

In questo argomento vengono forniti esempi di policy basate su identità in cui un amministratore account può collegare policy di autorizzazione a identità IAM, ovvero utenti, gruppi e ruoli.

Important

In primo luogo, è consigliabile leggere gli argomenti in cui vengono spiegati i concetti di base e le opzioni per gestire l'accesso alle risorse di Amazon ElastiCache. Per ulteriori

informazioni, consulta [Panoramica della gestione delle autorizzazioni di accesso alle risorse di ElastiCache](#).

In questa sezione vengono trattati gli argomenti seguenti:

- [Policy gestite da AWS per Amazon ElastiCache](#)
- [Esempi di policy gestite dal cliente](#)

Di seguito viene illustrato un esempio di policy di autorizzazione.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowClusterPermissions",
    "Effect": "Allow",
    "Action": [
      "elasticache:CreateServerlessCache",
      "elasticache:CreateCacheCluster",
      "elasticache:DescribeServerlessCaches",
      "elasticache:DescribeCacheClusters",
      "elasticache:ModifyServerlessCache",
      "elasticache:ModifyCacheCluster"
    ],
    "Resource": "*"
  },
  {
    "Sid": "AllowUserToPassRole",
    "Effect": "Allow",
    "Action": [ "iam:PassRole" ],
    "Resource": "arn:aws:iam::123456789012:role/EC2-roles-for-cluster"
  }
  ]
}
```

La policy include due dichiarazioni:

- La prima istruzione assegna le autorizzazioni per le operazioni di Amazon ElastiCache (`elasticache:Create*`, `elasticache:Describe*`, `elasticache:Modify*`)

- La seconda istruzione concede le autorizzazioni per l'operazione IAM (`iam:PassRole`) sul nome del ruolo IAM specificato alla fine del valore `Resource`.

La policy non specifica l'elemento `Principal` poiché in una policy basata su identità l'entità che ottiene l'autorizzazione non viene specificata. Quando si collega una policy a un utente, quest'ultimo è l'entità implicita. Quando colleghi una policy di autorizzazioni a un ruolo IAM, il principale identificato nella policy di attendibilità del ruolo ottiene le autorizzazioni.

Per una tabella che mostra tutte le operazioni API di Amazon ElastiCache e le risorse a cui si applicano, consulta [Autorizzazioni API di ElastiCache: riferimento a operazioni, risorse e condizioni](#).

Esempi di policy gestite dal cliente

Se non si utilizza una policy predefinita e si sceglie di utilizzare una policy gestita in modo personalizzato, assicurarsi di trovarsi in una delle due seguenti situazioni. O si dispone delle autorizzazioni per richiamare `iam:createServiceLinkedRole` (Per ulteriori informazioni, consulta [Esempio 4: assegnazione a un utente dell'autorizzazione a chiamare l'API IAM `CreateServiceLinkedRole`](#)). Oppure è stato creato un ruolo di ElastiCache legato al servizio.

Se combinate con le autorizzazioni minime necessarie per utilizzare la console di Amazon ElastiCache, le policy esemplificative in questa sessione concedono ulteriori autorizzazioni. Tali esempi sono, inoltre, pertinenti alle SDK AWS e alla AWS CLI.

Per istruzioni su come impostare gruppi e utenti IAM, consulta [Creazione del primo utente e gruppo di amministratori IAM](#) nella Guida per l'utente di IAM.

Important

Testa sempre in modo approfondito le Policy IAM prima di avvalertene in fase di produzione. Alcune operazioni di ElastiCache, apparentemente semplici, in realtà possono necessitare di altre operazioni di supporto, quando si utilizza la console di ElastiCache. Ad esempio, `elasticache:CreateCacheCluster` concede le autorizzazioni necessarie per creare cluster di cache di ElastiCache. Tuttavia, per portare a termine questa operazione, la console di ElastiCache si avvale di una serie di operazioni `Describe` e `List` per compilare gli elenchi di console.

Esempi

- [Esempio 1: assegnazione a un utente dell'accesso in sola lettura alle risorse ElastiCache](#)

- [Esempio 2: assegnazione a un utente dell'autorizzazione a eseguire operazioni di amministrazione comuni del sistema ElastiCache](#)
- [Esempio 3: assegnazione a un utente dell'accesso a tutte le operazioni dell'API ElastiCache](#)
- [Esempio 4: assegnazione a un utente dell'autorizzazione a chiamare l'API IAM `CreateServiceLinkedRole`](#)

Esempio 1: assegnazione a un utente dell'accesso in sola lettura alle risorse ElastiCache

La policy seguente concede le autorizzazioni necessarie per le operazioni di ElastiCache che permettono all'utente di consultare le risorse in elenco. In genere, si collega questo tipo di policy di autorizzazione a un gruppo di gestori.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "ECReadOnly",
    "Effect": "Allow",
    "Action": [
      "elasticache:Describe*",
      "elasticache:List*"
    ],
    "Resource": "*"
  }
]
```

Esempio 2: assegnazione a un utente dell'autorizzazione a eseguire operazioni di amministrazione comuni del sistema ElastiCache

Le attività comuni dell'amministratore di sistema includono la modifica delle risorse. Un amministratore di sistema potrebbe, inoltre, voler reperire informazioni sugli eventi di ElastiCache. La policy seguente concede a un utente le autorizzazioni necessarie per eseguire le operazioni di ElastiCache correlate alle più comuni attività degli amministratori di sistema, già citate. In genere, si collega questo tipo di policy di autorizzazione al gruppo degli amministratori di sistema.

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



```

        "elasticache:Modify*",
        "elasticache:Describe*",
        "elasticache:ResetCacheParameterGroup"
    ],
    "Resource": "*"
}
]
}

```

Esempio 3: assegnazione a un utente dell'accesso a tutte le operazioni dell'API ElastiCache

La policy seguente permette a un utente di accedere a tutte le operazioni di ElastiCache. Consigliamo di concedere questo tipo di policy di autorizzazione solo a un utente amministratore.

```

{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "ECAAllowAll",
    "Effect": "Allow",
    "Action": [
      "elasticache:*"
    ],
    "Resource": "*"
  }
]
}

```

Esempio 4: assegnazione a un utente dell'autorizzazione a chiamare l'API IAM CreateServiceLinkedRole

La policy seguente permette a un utente di chiamare l'API IAM CreateServiceLinkedRole . Consigliamo di concedere questo tipo di policy di autorizzazione all'utente che invoca le operazioni mutative di ElastiCache.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CreateSLRAllows",
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ]
    }
  ]
}

```

```
    ],
    "Resource": "*",
    "Condition": {
      "StringLike": {
        "iam:AWSServiceName": "elasticache.amazonaws.com"
      }
    }
  }
]
```

Autorizzazioni a livello di risorsa

È possibile limitare la portata delle autorizzazioni specificando le risorse in una policy IAM. Molte operazioni dell'API ElastiCache supportano un tipo di risorsa che varia a seconda del comportamento dell'operazione. Ogni dichiarazione di policy IAM concede l'autorizzazione a un'operazione eseguita su una risorsa. Quando l'operazione non agisce su una risorsa designata oppure quando concedi l'autorizzazione per eseguire l'operazione su tutte le risorse, il valore della risorsa nella policy è un carattere jolly (*). Per molte operazioni API è possibile limitare le risorse che un utente può modificare specificando l'Amazon Resource Name (ARN) di una risorsa o un modello ARN che soddisfa più risorse. Per limitare le autorizzazioni in base alla risorsa, specifica la risorsa in base all'ARN.

Per visualizzare un elenco di tipi di risorse ElastiCache e i relativi ARN, consulta [Tipi di risorsa definiti da Amazon ElastiCache](#) nella Guida di riferimento per l'autorizzazione del servizio. Per informazioni sulle operazioni con cui è possibile specificare l'ARN di ogni risorsa, consulta [Operazioni definite da Amazon ElastiCache](#).

Esempi

- [Esempio 1: consentire a un utente l'accesso completo a tipi di risorse di ElastiCache specifiche](#)
- [Esempio 2: rifiutare a un utente l'accesso a una cache serverless](#)

Esempio 1: consentire a un utente l'accesso completo a tipi di risorse di ElastiCache specifiche

La policy seguente consente esplicitamente tutte le risorse di tipo cache serverless.

```
{
  "Sid": "Example1",
  "Effect": "Allow",
  "Action": "elasticache:*",
  "Resource": [
```

```
    "arn:aws:elasticache:us-east-1:account-id:serverlesscache:*"
  ]
}
```

Esempio 2: rifiutare a un utente l'accesso a una cache serverless

Nell'esempio seguente viene negato esplicitamente l'accesso a una determinata cache serverless.

```
{
  "Sid": "Example2",
  "Effect": "Deny",
  "Action": "elasticache:*",
  "Resource": [
    "arn:aws:elasticache:us-east-1:account-id:serverlesscache:name"
  ]
}
```

Utilizzo delle chiavi di condizione

Puoi specificare le condizioni che determinano il modo in cui una policy IAM viene applicata. Su ElastiCache è possibile utilizzare l'elemento `Condition` di una policy JSON per confrontare le chiavi della richiesta con i valori chiave specificati nella policy. Per ulteriori informazioni, consulta [elementi della policy IAM JSON: condizione](#).

Per un elenco delle chiavi di condizione di ElastiCache, consulta [Chiavi di condizione per Amazon ElastiCache](#) nella Guida di riferimento per l'autorizzazione del servizio.

Per un elenco delle chiavi di condizione globali, consulta [Chiavi di contesto delle condizioni globaliAWS](#).

Specifica delle condizioni: Uso delle chiavi di condizione

Per implementare un controllo particolareggiato, scrivi una policy relativa alle autorizzazioni IAM che specifichi le condizioni per controllare un set di singoli parametri su determinate richieste. Quindi la policy viene applicata agli utenti, ai gruppi o ai ruoli IAM creati utilizzando la console IAM.

Per applicare una condizione, aggiungere le informazioni sulla condizione all'istruzione della policy IAM. Nell'esempio seguente viene specificata la condizione secondo la quale qualsiasi cluster di cache progettato autonomamente avrà nodi di tipo `cache.r5.large`.

```
{
  "Version": "2012-10-17",
```

```
"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "elasticache:CreateCacheCluster"
    ],
    "Resource": [
      "arn:aws:elasticache:*:*:parametergroup:*",
      "arn:aws:elasticache:*:*:subnetgroup:*"
    ]
  },
  {
    "Effect": "Allow",
    "Action": [
      "elasticache:CreateCacheCluster"
    ],
    "Resource": [
      "arn:aws:elasticache:*:*:cluster:*"
    ],
    "Condition": {
      "StringEquals": {
        "elasticache:CacheNodeType": [
          "cache.r5.large"
        ]
      }
    }
  }
]
```

Per maggiori informazioni, consulta [Esempi di policy di controllo dell'accesso basate su tag](#).

Per ulteriori informazioni sull'utilizzo degli operatori delle condizioni di policy, consulta [Autorizzazioni API di ElastiCache: riferimento a operazioni, risorse e condizioni](#).

Policy di esempio: Utilizzo di condizioni per il controllo granulare dei parametri

In questa sezione vengono illustrate le policy di esempio per implementare il controllo granulare degli accessi sui parametri di ElastiCache elencati in precedenza.

1. `elasticache:MaximumDataStorage`: specifica l'archiviazione di dati massima di una cache serverless. Utilizzando le condizioni fornite, il cliente non può creare cache in grado di archiviare più di una quantità specifica di dati.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowDependentResources",
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscachesnapshot:*",
        "arn:aws:elasticache:*:*:snapshot:*",
        "arn:aws:elasticache:*:*:usergroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscache:*"
      ],
      "Condition": {
        "NumericLessThanEquals": {
          "elasticache:MaximumDataStorage": "30"
        },
        "StringEquals": {
          "elasticache:DataStorageUnit": "GB"
        }
      }
    }
  ]
}

```

2. `elasticache:MaximumECPUPerSecond`: specifica il valore massimo di ECPU al secondo di una cache serverless. Utilizzando le condizioni fornite, il cliente non può creare cache in grado di eseguire più di un numero specifico di ECPU al secondo.

```

{
  "Version": "2012-10-17",
  "Statement": [

```

```

    {
      "Sid": "AllowDependentResources",
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscachesnapshot:*",
        "arn:aws:elasticache:*:*:snapshot:*",
        "arn:aws:elasticache:*:*:usergroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscache:*"
      ],
      "Condition": {
        "NumericLessThanEquals": {
          "elasticache:MaximumECPUPerSecond": "100000"
        }
      }
    }
  ]
}

```

3. `elasticache:CacheNodeType`: Specificare quali `NodeType` un utente può creare. Utilizzando le condizioni fornite, il cliente può specificare un valore singolo o un valore di intervallo per un tipo di nodo.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",

```

```

        "arn:aws:elasticache:*:*:subnetgroup:*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "elasticache:CreateCacheCluster"
    ],
    "Resource": [
        "arn:aws:elasticache:*:*:cluster:*"
    ],
    "Condition": {
        "StringEquals": {
            "elasticache:CacheNodeType": [
                "cache.t2.micro",
                "cache.t2.medium"
            ]
        }
    }
}
]
}

```

4. elasticache:EngineVersion: Specificare l'utilizzo della versione del motore 1.6.6

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "elasticache:CreateCacheCluster"
            ],
            "Resource": [
                "arn:aws:elasticache:*:*:parametergroup:*",
                "arn:aws:elasticache:*:*:subnetgroup:*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "elasticache:CreateCacheCluster"
            ]
        }
    ]
}

```

```

    ],
    "Resource": [
        "arn:aws:elasticache:*:*:cluster:*"
    ],
    "Condition": {
        "StringEquals": {
            "elasticache:EngineVersion": "1.6.6"
        }
    }
}
]
}

```

5. `elasticache:KmsKeyId`: Specificare l'utilizzo delle chiavi KMS AWS gestite dal cliente.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowDependentResources",
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscachesnapshot:*",
        "arn:aws:elasticache:*:*:snapshot:*",
        "arn:aws:elasticache:*:*:usergroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateServerlessCache"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:serverlesscache:*"
      ],
      "Condition": {
        "StringEquals": {
            "elasticache:KmsKeyId": "my-key"
        }
      }
    }
  ]
}

```



```
]
}
```

6. `elasticache:CacheParameterGroupName`: Specificare un gruppo di parametri non predefinito con parametri specifici di un'organizzazione sui tuoi cluster. È inoltre possibile specificare un modello di denominazione per i gruppi di parametri o eliminare blocchi su un nome di gruppo di parametri specifico. Di seguito è riportato un esempio che vincola l'utilizzo di solo "my-org-param-group".

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*"
      ],
      "Condition": {
        "StringEquals": {
          "elasticache:CacheParameterGroupName": "my-org-param-group"
        }
      }
    }
  ]
}
```

7. `elasticache:CreateCacheCluster`: rifiuto dell'azione `CreateCacheCluster` se il tag della richiesta `Project` manca o è diverso da `Dev`, `QA` o `Prod`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*",
        "arn:aws:elasticache:*:*:securitygroup:*",
        "arn:aws:elasticache:*:*:replicationgroup:*"
      ]
    },
    {
      "Effect": "Deny",
      "Action": [
        "elasticache:CreateCacheCluster"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*"
      ],
      "Condition": {
        "Null": {
          "aws:RequestTag/Project": "true"
        }
      }
    }
  ],
  {
    "Effect": "Allow",
    "Action": [
      "elasticache:CreateCacheCluster",
      "elasticache:AddTagsToResource"
    ],
    "Resource": "arn:aws:elasticache:*:*:cluster:*",
    "Condition": {
      "StringEquals": {
        "aws:RequestTag/Project": [
          "Dev",
          "Prod",
          "QA"
        ]
      }
    }
  ]
}
```

```

    }
  }
}
]
}

```

8. `elasticache:CacheNodeType`: consente `CreateCacheCluster` con `cacheNodeType` `cache.r5.large` o `cache.r6g.4xlarge` e tag `Project=XYZ`.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:parametergroup:*",
        "arn:aws:elasticache:*:*:subnetgroup:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:CreateCacheCluster"
      ],
      "Resource": [
        "arn:aws:elasticache:*:*:cluster:*"
      ],
      "Condition": {
        "StringEqualsIfExists": {
          "elasticache:CacheNodeType": [
            "cache.r5.large",
            "cache.r6g.4xlarge"
          ]
        },
        "StringEquals": {
          "aws:RequestTag/Project": "XYZ"
        }
      }
    }
  ]
}

```

}

Note

Quando si creano policy per applicare tag e altre chiavi di condizione insieme, il condizionale `IfExists` può essere richiesto su elementi di condizione chiave a causa dei requisiti della policy `elasticache:AddTagsToResource` aggiuntivi per le richieste di creazione con il parametro `--tags`.

Utilizzo di ruoli collegati ai servizi per Amazon ElastiCache

Amazon ElastiCache utilizza [ruoli collegati ai servizi](#) di AWS Identity and Access Management (IAM). Un ruolo collegato ai servizi è un tipo univoco di ruolo IAM collegato direttamente a un servizio AWS, come Amazon ElastiCache. I ruoli collegati al servizio di Amazon ElastiCache sono predefiniti da Amazon ElastiCache. Includono tutte le autorizzazioni necessarie al servizio per richiamare servizi AWS per conto dei cluster.

Un ruolo collegato ai servizi semplifica la configurazione di Amazon ElastiCache perché permette di evitare l'aggiunta manuale delle autorizzazioni necessarie. I ruoli sono già presenti nel tuo account AWS ma sono collegati ai casi d'uso di Amazon ElastiCache e dispongono di autorizzazioni predefinite. Solo Amazon ElastiCache può assumere questi ruoli e solo questi ruoli possono usare la policy delle autorizzazioni predefinite. È possibile eliminare i ruoli solo dopo aver eliminato le risorse correlate. Questa procedura protegge le risorse di Amazon ElastiCache poiché impedisce la rimozione involontaria delle necessarie autorizzazioni di accesso alle risorse.

Per informazioni sugli altri servizi che supportano i ruoli collegati ai servizi, consulta [Servizi AWS che funzionano con IAM](#) e cerca i servizi che riportano Sì nella colonna Ruolo associato ai servizi. Scegliere un link Yes (Sì) per visualizzare la documentazione relativa al ruolo collegato ai servizi per tale servizio.

Indice

- [Autorizzazioni del ruolo collegato ai servizi per Amazon ElastiCache](#)
 - [Autorizzazioni per creare un ruolo collegato ai servizi](#)
- [Creazione di un ruolo collegato ai servizi \(IAM\)](#)
 - [Creazione di un ruolo collegato ai servizi \(Console di IAM\)](#)

- [Creazione di un ruolo collegato ai servizi \(CLI di IAM\)](#)
- [Creazione di un ruolo collegato ai servizi \(API di IAM\)](#)
- [Modifica della descrizione di un ruolo collegato ai servizi per Amazon ElastiCache](#)
 - [Modifica della descrizione di un ruolo collegato ai servizi \(console di IAM\)](#)
 - [Modifica della descrizione di un ruolo collegato ai servizi \(CLI di IAM\)](#)
 - [Modifica della descrizione di un ruolo collegato ai servizi \(API di IAM\)](#)
- [Eliminazione di un ruolo collegato ai servizi per Amazon ElastiCache](#)
 - [Pulizia di un ruolo collegato ai servizi](#)
 - [Eliminazione di un ruolo collegato ai servizi \(console di IAM\)](#)
 - [Eliminazione di un ruolo collegato ai servizi \(CLI di IAM\)](#)
 - [Eliminazione di un ruolo collegato ai servizi \(API di IAM\)](#)

Autorizzazioni del ruolo collegato ai servizi per Amazon ElastiCache

Autorizzazioni per creare un ruolo collegato ai servizi

Per consentire a un'entità IAM di creare il ruolo collegato ai servizi `AWSServiceRoleForElastiCache`

Aggiungi la seguente istruzione di policy alle autorizzazioni per l'entità IAM.

```
{
  "Effect": "Allow",
  "Action": [
    "iam:CreateServiceLinkedRole",
    "iam:PutRolePolicy"
  ],
  "Resource": "arn:aws:iam::*:role/aws-service-role/elasticache.amazonaws.com/
AWSServiceRoleForElastiCache*",
  "Condition": {"StringLike": {"iam:AWSserviceName": "elasticache.amazonaws.com"}}
}
```

Per consentire a un'entità IAM di eliminare il ruolo collegato ai servizi `AWSServiceRoleForElastiCache`

Aggiungi la seguente istruzione di policy alle autorizzazioni per l'entità IAM.

```
{
```

```
"Effect": "Allow",
"Action": [
    "iam:DeleteServiceLinkedRole",
    "iam:GetServiceLinkedRoleDeletionStatus"
],
"Resource": "arn:aws:iam:*:role/aws-service-role/elasticache.amazonaws.com/
AWSServiceRoleForElastiCache*",
"Condition": {"StringLike": {"iam:AWSServiceName": "elasticache.amazonaws.com"}}
}
```

In alternativa, è possibile utilizzare una policy gestita da AWS per fornire l'accesso completo ad Amazon ElastiCache.

Creazione di un ruolo collegato ai servizi (IAM)

È possibile creare un ruolo collegato ai servizi utilizzando la console di IAM, la CLI o l'API.

Creazione di un ruolo collegato ai servizi (Console di IAM)

Puoi utilizzare la console IAM per creare un ruolo collegato ai servizi.

Come creare un ruolo collegato ai servizi (console)

1. Accedi alla AWS Management Console e apri la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.
2. Nel pannello di navigazione della console IAM seleziona Roles (Ruoli). Quindi seleziona Create new role (Crea nuovo ruolo).
3. In Select type of trusted entity (Seleziona tipo di entità attendibile), scegli Service AWS.
4. In Or select a service to view its use cases (Oppure seleziona un servizio per visualizzarne i casi d'uso), scegli ElastiCache.
5. Scegli Successivo: Autorizzazioni.
6. In Policy name (Nome policy), si noti che `ElastiCacheServiceRolePolicy` è necessario per questo ruolo. Scegli Successivo: Tag.
7. Si noti che i tag non sono supportati per i ruoli collegati al servizio. Scegliere Next:Review (Successivo:Rivedi).
8. (Facoltativo) In Role description (Descrizione ruolo) modifica la descrizione per il nuovo ruolo collegato ai servizi.
9. Rivedere il ruolo e scegliere Crea ruolo.

Creazione di un ruolo collegato ai servizi (CLI di IAM)

È possibile utilizzare le operazioni di IAM da AWS Command Line Interface per creare un ruolo collegato ai servizi. Questo ruolo può includere la policy di attendibilità e le policy inline che il servizio richiede per assumere il ruolo.

Per creare un ruolo collegato ai servizi (CLI)

Attenersi alle operazioni seguenti:

```
$ aws iam create-service-linked-role --aws-service-name elasticache.amazonaws.com
```

Creazione di un ruolo collegato ai servizi (API di IAM)

È possibile utilizzare l'API di IAM per creare un ruolo collegato ai servizi. Questo ruolo può contenere la policy di attendibilità e le policy inline che il servizio richiede per assumere il ruolo.

Per creare un ruolo collegato ai servizi (API)

Utilizzare la chiamata API [CreateServiceLinkedRole](#). Nella richiesta, specificare un nome del servizio di `elasticache.amazonaws.com`.

Modifica della descrizione di un ruolo collegato ai servizi per Amazon ElastiCache

Amazon ElastiCache non consente di modificare il ruolo collegato ai servizi `AWSServiceRoleForElastiCache`. Dopo aver creato un ruolo collegato al servizio, non potrai modificarne il nome perché varie entità potrebbero farvi riferimento. È possibile tuttavia modificarne la descrizione utilizzando IAM.

Modifica della descrizione di un ruolo collegato ai servizi (console di IAM)

È possibile utilizzare la console di IAM per modificare la descrizione di un ruolo collegato ai servizi.

Per modificare la descrizione di un ruolo collegato ai servizi (console)

1. Nel pannello di navigazione della console IAM seleziona Roles (Ruoli).
2. Scegliere il nome del ruolo da modificare.
3. Nella parte destra di Role description (Descrizione ruolo), scegliere Edit (Modifica).
4. Digita una nuova descrizione nella casella e scegli Save (Salva).

Modifica della descrizione di un ruolo collegato ai servizi (CLI di IAM)

È possibile utilizzare le operazioni di IAM dalla AWS Command Line Interface per modificare la descrizione di un ruolo collegato ai servizi.

Per modificare la descrizione di un ruolo collegato ai servizi (CLI)

1. (Facoltativo) Per visualizzare la descrizione corrente di un ruolo, utilizzare la AWS CLI per l'operazione di IAM [get-role](#).

Example

```
$ aws iam get-role --role-name AWSServiceRoleForElastiCache
```

Utilizzare il nome del ruolo, non l'ARN, per fare riferimento ai ruoli con le operazioni CLI. Ad esempio, per fare riferimento a un ruolo il cui ARN è `arn:aws:iam::123456789012:role/myrole`, puoi usare **myrole**.

2. Per aggiornare la descrizione di un ruolo collegato ai servizi, utilizza la AWS CLI per l'operazione di IAM [update-role-description](#).

Per Linux, macOS o Unix:

```
$ aws iam update-role-description \  
  --role-name AWSServiceRoleForElastiCache \  
  --description "new description"
```

Per Windows:

```
$ aws iam update-role-description ^  
  --role-name AWSServiceRoleForElastiCache ^  
  --description "new description"
```

Modifica della descrizione di un ruolo collegato ai servizi (API di IAM)

È possibile utilizzare l'API di IAM per modificare la descrizione di un ruolo collegato ai servizi.

Per modificare la descrizione di un ruolo collegato ai servizi (API)

1. (Facoltativo) Per visualizzare la descrizione corrente di un ruolo, utilizzare l'operazione API di IAM [GetRole](#).

Example

```
https://iam.amazonaws.com/  
?Action=GetRole  
&RoleName=AWSServiceRoleForElastiCache  
&Version=2010-05-08  
&AUTHPARAMS
```

2. Per aggiornare la descrizione di un ruolo, utilizzare l'operazione API di IAM [UpdateRoleDescription](#).

Example

```
https://iam.amazonaws.com/  
?Action=UpdateRoleDescription  
&RoleName=AWSServiceRoleForElastiCache  
&Version=2010-05-08  
&Description="New description"
```

Eliminazione di un ruolo collegato ai servizi per Amazon ElastiCache

Se non è più necessario utilizzare una caratteristica o un servizio che richiede un ruolo collegato ai servizi, ti consigliamo di eliminare il ruolo. In questo modo non sarà più presente un'entità non utilizzata che non viene monitorata e gestita attivamente. Tuttavia, è necessario effettuare la pulizia delle risorse associate al ruolo collegato ai servizi prima di poterlo eliminare.

Amazon ElastiCache non elimina il ruolo collegato ai servizi per tuo conto.

Pulizia di un ruolo collegato ai servizi

Prima di utilizzare IAM per eliminare un ruolo collegato ai servizi, verifica innanzitutto che il ruolo non abbia risorse o cluster associati al ruolo.

Per verificare se il ruolo collegato ai servizi dispone di una sessione attiva nella console IAM

1. Accedi alla AWS Management Console e apri la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.
2. Nel pannello di navigazione della console IAM seleziona Roles (Ruoli). Quindi, scegli il nome (non la casella di controllo) del ruolo AWSServiceRoleForElastiCache
3. Nella pagina Summary (Riepilogo) per il ruolo selezionato, scegliere la scheda Access Advisor (Consulente accessi).
4. Nella scheda Access Advisor (Consulente accessi), esamina l'attività recente per il ruolo collegato ai servizi.

Per eliminare le risorse Amazon ElastiCache che richiedono AWSServiceRoleForElastiCache

- Per eliminare un cluster, consultare i seguenti argomenti:
 - [Utilizzo di AWS Management Console](#)
 - [Utilizzo di AWS CLI](#)
 - [Utilizzo dell'API ElastiCache](#)

Eliminazione di un ruolo collegato ai servizi (console di IAM)

È possibile utilizzare la console IAM per eliminare un ruolo collegato ai servizi.

Per eliminare un ruolo collegato ai servizi (console)

1. Accedi alla AWS Management Console e apri la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.
2. Nel pannello di navigazione della console IAM seleziona Roles (Ruoli). Quindi, seleziona la casella di controllo accanto al nome del ruolo che desideri eliminare, non il nome o la riga stessa.
3. In operazioni Role (Ruolo) nella parte superiore della pagina, seleziona Delete (Elimina) ruolo.
4. Nella finestra di dialogo di conferma controlla i dati relativi all'ultimo accesso ai servizi, che indicano quando ognuno dei ruoli selezionati ha effettuato l'accesso a un servizio AWS. In questo modo potrai verificare se il ruolo è attualmente attivo. Se desideri procedere, seleziona Yes, Delete (Sì, elimina) per richiedere l'eliminazione del ruolo collegato ai servizi.
5. Controlla le notifiche della console IAM per monitorare lo stato dell'eliminazione del ruolo collegato ai servizi. Poiché l'eliminazione del ruolo collegato ai servizi IAM è asincrona, una

volta richiesta l'eliminazione del ruolo, il task di eliminazione può essere eseguito correttamente o meno. Se il task non viene eseguito correttamente, puoi scegliere View details (Visualizza dettagli) o View Resources (Visualizza risorse) dalle notifiche per capire perché l'eliminazione non è stata effettuata.

Eliminazione di un ruolo collegato ai servizi (CLI di IAM)

È possibile utilizzare le operazioni di IAM dalla AWS Command Line Interface per eliminare un ruolo collegato ai servizi.

Per eliminare un ruolo collegato ai servizi (CLI)

1. Se non conosci il nome del ruolo collegato ai servizi da eliminare, inserisci il comando seguente: Questo comando elenca i ruoli e i relativi Amazon Resource Name (ARN) nel tuo account.

```
$ aws iam get-role --role-name role-name
```

Utilizzare il nome del ruolo, non l'ARN, per fare riferimento ai ruoli con le operazioni CLI. Ad esempio, per fare riferimento a un ruolo il cui ARN è `arn:aws:iam::123456789012:role/myrole`, puoi usare **myrole**.

2. Poiché un ruolo collegato ai servizi non può essere eliminato se è in uso o se a esso sono associate delle risorse, occorre inviare una richiesta di eliminazione. Se queste condizioni non sono soddisfatte, la richiesta può essere rifiutata. Acquisisci il valore di `deletion-task-id` dalla risposta per controllare lo stato del task di eliminazione. Per inviare una richiesta di eliminazione per un ruolo collegato ai servizi, inserire quanto segue:

```
$ aws iam delete-service-linked-role --role-name role-name
```

3. Inserire quanto segue per verificare lo stato del processo di eliminazione:

```
$ aws iam get-service-linked-role-deletion-status --deletion-task-id deletion-task-id
```

Lo stato di un task di eliminazione può essere NOT_STARTED, IN_PROGRESS, SUCCEEDED o FAILED. Se l'eliminazione non viene eseguita correttamente, la chiamata restituisce il motivo dell'errore per consentire all'utente di risolvere il problema.

Eliminazione di un ruolo collegato ai servizi (API di IAM)

È possibile utilizzare l'API di IAM; per eliminare un ruolo collegato ai servizi.

Per eliminare un ruolo collegato ai servizi (API)

1. Per inviare una richiesta di eliminazione per un ruolo collegato ai servizi, chiamare [DeleteServiceLinkedRole](#). Nella richiesta, specifica il nome del ruolo.

Poiché un ruolo collegato ai servizi non può essere eliminato se è in uso o se a esso sono associate delle risorse, occorre inviare una richiesta di eliminazione. Se queste condizioni non sono soddisfatte, la richiesta può essere rifiutata. Acquisisci il valore di `DeletionTaskId` dalla risposta per controllare lo stato del task di eliminazione.

2. Chiamare [GetServiceLinkedRoleDeletionStatus](#) per controllare lo stato dell'eliminazione. Nella richiesta, specificare il `DeletionTaskId`.

Lo stato di un task di eliminazione può essere `NOT_STARTED`, `IN_PROGRESS`, `SUCCEEDED` o `FAILED`. Se l'eliminazione non viene eseguita correttamente, la chiamata restituisce il motivo dell'errore per consentire all'utente di risolvere il problema.

Autorizzazioni API di ElastiCache: riferimento a operazioni, risorse e condizioni

Utilizzare la seguente tabella come riferimento quando si configura il [controllo dell'accesso](#) e si scrivono policy di autorizzazione da collegare a policy IAM (policy basate sulle identità o sulle risorse). La tabella seguente elenca le operazioni API di Amazon ElastiCache e le operazioni corrispondenti cui hai concesso le autorizzazioni di esecuzione dell'operazione. Puoi specificare le operazioni nel campo `Action` della policy e il valore di una risorsa nel campo `Resource` della policy. Se non diversamente indicato, la risorsa è obbligatoria. Alcuni campi includono sia una risorsa obbligatoria che risorse facoltative. Quando non è presente alcuna risorsa ARN, la risorsa nella policy è un carattere jolly (*).

Puoi usare le chiavi di condizione nelle policy di ElastiCache per esprimere le condizioni. Per visualizzare un elenco delle chiavi di condizione specifiche di Elasticache, insieme alle operazioni e ai tipi di risorse a cui si applicano, consulta [Utilizzo delle chiavi di condizione](#). Per l'elenco completo delle chiavi AWS, consulta [Chiavi di contesto e condizione globali di AWS](#) nella Guida per l'utente di IAM.

Note

Per specificare un'operazione, utilizza il prefisso `elasticache:` seguito dal nome dell'operazione API (ad esempio, `elasticache:DescribeCacheClusters`).

Per visualizzare un elenco di azioni ElastiCache, consulta [Operazioni definite da Amazon ElastiCache](#) nella Guida di riferimento per l'autorizzazione del servizio.

Convalida della conformità per Amazon ElastiCache


Revisori di terza parte valutano la sicurezza e la conformità di AWS come parte di diversi programmi di conformità di AWS, come SOC, PCI, FedRAMP, HIPAA e altri.

Per sapere se il Servizio AWS è coperto da programmi di conformità specifici, consulta i [Servizi AWS coperti dal programma di conformità](#) e scegli il programma di conformità desiderato. Per informazioni generali, consulta [Programmi per la conformità di AWS](#).

È possibile scaricare i report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta [Download di report in AWS Artifact](#).

La responsabilità di conformità durante l'utilizzo dei Servizi AWS è determinata dalla riservatezza dei dati, dagli obiettivi di conformità dell'azienda e dalle normative vigenti. Per semplificare il rispetto della conformità, AWS mette a disposizione le seguenti risorse:

- [Guide Quick Start per la sicurezza e conformità](#): queste guide all'implementazione illustrano considerazioni relative all'architettura e forniscono la procedura per l'implementazione di ambienti di base su AWS incentrati sulla sicurezza e sulla conformità.
- [Architetture per la sicurezza e la conformità HIPAA su Amazon Web Services](#): questo whitepaper descrive come le aziende possono utilizzare AWS per creare applicazioni conformi alla normativa HIPAA.

 Note

Non tutti i Servizi AWS sono conformi ai requisiti HIPAA. Per ulteriori informazioni, consulta la sezione [Riferimenti sui servizi conformi ai requisiti HIPAA](#).

- [Risorse per la conformità AWS](#): una raccolta di cartelle di lavoro e guide suddivise per settore e area geografica.
- [AWS Guide alla conformità dei clienti](#): comprendi il modello di responsabilità condivisa attraverso la lente della conformità. Le guide riassumono le migliori pratiche per la protezione Servizi AWS e mappano le linee guida per i controlli di sicurezza su più framework (tra cui il National Institute of Standards and Technology (NIST), il Payment Card Industry Security Standards Council (PCI) e l'International Organization for Standardization (ISO)).
- [Valutazione delle risorse con le regole](#) nella Guida per gli sviluppatori di AWS Config: il servizio AWS Config valuta il livello di conformità delle configurazioni delle risorse con pratiche interne, linee guida e regolamenti.
- [AWS Security Hub](#): questo Servizio AWS fornisce una visione completa dello stato di sicurezza all'interno di AWS. La Centrale di sicurezza utilizza i controlli di sicurezza per valutare le risorse AWS e verificare la conformità agli standard e alle best practice del settore della sicurezza. Per un elenco dei servizi e dei controlli supportati, consulta la pagina [Documentazione di riferimento sui controlli della Centrale di sicurezza](#).
- [AWS Audit Manager](#): questo Servizio AWS aiuta a verificare continuamente l'utilizzo di AWS per semplificare la gestione dei rischi e della conformità alle normative e agli standard di settore.

Ulteriori informazioni

Per informazioni generali sulla conformità a Cloud AWS, consulta i seguenti documenti:

- [Endpoint FIPS per servizio](#)
- [Aggiornamenti del servizio in ElastiCache](#)
- [Conformità di sicurezza nel cloud AWS](#)
- [Modello di responsabilità condivisa](#)
- [AWSProgramma di Conformità PCI DSS](#)

Resilienza in Amazon ElastiCache

L'infrastruttura globale di AWS è basata su Regioni e zone di disponibilità AWS. AWS Le Regioni forniscono più zone di disponibilità fisicamente separate e isolate che sono connesse tramite reti altamente ridondanti, a bassa latenza e velocità effettiva elevata. Con le zone di disponibilità, è possibile progettare e gestire le applicazioni e database che eseguono il failover automatico tra zone di disponibilità senza interruzioni. Le zone di disponibilità sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture a data center singolo o multiplo.

Per ulteriori informazioni sulle Regioni AWS e sulle zone di disponibilit, consulta [Infrastruttura globale di AWS](#).

Oltre all'infrastruttura globale AWS, Amazon ElastiCache offre numerose caratteristiche per supportare la resilienza dei dati e le esigenze di backup.

Argomenti

- [Limitazione dell'impatto degli errori](#)

Limitazione dell'impatto degli errori

Quando pianifichi l' ElastiCache implementazione di Amazon, devi pianificare in modo che gli errori abbiano un impatto minimo sull'applicazione e sui dati. Questa sezione, organizzata in più argomenti, illustra cosa fare per proteggere l'applicazione e i dati in caso di errori.

Argomenti

- [Limitazione degli errori con Memcached in esecuzione](#)
- [Raccomandazioni](#)

Limitazione degli errori con Memcached in esecuzione

Per minimizzare l'impatto degli errori con un motore Memcached in esecuzione, è possibile avvalersi delle seguenti soluzioni. Esistono due tipi di errori gestibili nei piani di mitigazione, gli errori dei nodi e quelli delle zone di disponibilità.

Limitazione degli errori dei nodi

Le cache serverless mitigano automaticamente gli errori dei nodi con un'architettura multi-AZ replicata in modo che gli errori dei nodi siano trasparenti per l'applicazione. Per minimizzare l'impatto dell'errore di un nodo in un cluster progettato autonomamente, occorre innanzitutto distribuire i dati memorizzati nella cache tra più nodi. Poiché i cluster progettati autonomamente non supportano la replica, un errore di nodo comporta in ogni caso una perdita di dati del cluster.

Quando crei un cluster Memcached, puoi crearlo con da 1 a 60 nodi o più su richiesta speciale. Ripartire i dati tra più nodi consente di arginare la perdita di dati se un nodo genera un errore. Se, ad esempio, i dati vengono distribuiti tra dieci nodi, ciascuno di questi includerà circa il 10% dei dati memorizzati nella cache. Pertanto, in caso di errore, la perdita da arginare con la creazione e il provisioning di un nodo sostitutivo sarà del 10%. Se gli stessi dati fossero memorizzati in 3 nodi più grandi, un errore di nodo comporterebbe la perdita di circa il 33% dei dati memorizzati nella cache.

[Se hai bisogno di più di 60 nodi in un cluster Memcached o più di 300 nodi in totale in una AWS regione, compila il modulo di richiesta di aumento del ElastiCache limite all'indirizzo https://aws.amazon.com/contact-us//.elasticache-node-limit-request](https://aws.amazon.com/contact-us//.elasticache-node-limit-request)

Per informazioni su come definire il numero di nodi in un cluster Memcached, consulta [Creazione di un cluster Memcached \(console\)](#).

Limitazione degli errori delle zone di disponibilità

Le cache serverless mitigano automaticamente gli errori delle zone di disponibilità con un'architettura multi-AZ replicata in modo che gli errori delle zone di disponibilità siano trasparenti per l'applicazione.

Per limitare l'impatto dell'errore di una zona di disponibilità in un cluster progettato autonomamente, posiziona i nodi nel maggior numero possibile di zone. Nell'improbabile caso di un errore di zona, si perderanno i dati memorizzati nella cache solo di tale zona e non di altre.

Perché impostare tanti nodi?

Se la mia regione ha solo tre zone di disponibilità, perché mi occorrono più di tre nodi, dal momento che, se una zona genera un errore, perdo comunque un terzo dei dati?

Ottima domanda. Esistono due tipi di errori di cui limitare gli effetti, gli errori dei nodi e quelli delle zone di disponibilità. Effettivamente, se i dati sono distribuiti su più zone di disponibilità, una delle quali genera un errore, si perdono solo i dati memorizzati in quest'ultima, indipendentemente dal numero di nodi in essere. Tuttavia, se è un nodo a generare l'errore, la presenza di più nodi riduce proporzionalmente la perdita di dati.

Non esiste una "formula magica" che consenta di determinare il numero di nodi da configurare in un cluster. È necessario valutare l'eventuale impatto della perdita di dati rispetto alla probabilità di errore e ai costi, per poi trarre le proprie conclusioni.

Per informazioni su come definire il numero di nodi in un cluster Memcached, consulta [Creazione di un cluster Memcached \(console\)](#).

Per ulteriori informazioni sulle regioni e sulle zone di disponibilità, consulta [Regioni e zone di disponibilità](#).

Raccomandazioni

Consigliamo di creare le cache serverless su cluster progettati autonomamente, in quanto si ottiene automaticamente una migliore tolleranza agli errori senza configurazioni aggiuntive. Quando si crea un cluster progettato autonomamente, tuttavia, è necessario pianificare due tipi di errori: errori di singoli nodi ed errori di zone di disponibilità. Il piano di mitigazione degli errori ideale consente di affrontare entrambe le tipologie di guasto.

Minimizzazione dell'impatto degli errori dei nodi

Quando si esegue Memcached e si ripartiscono i dati tra i nodi, più nodi si configurano, minore sarà la perdita di dati in caso di errore.

Minimizzare l'impatto degli errori delle zone di disponibilità

Per ridurre al minimo l'impatto dell'errore di una zona di disponibilità, consigliamo di ubicare i nodi nel maggior numero possibile di zone di disponibilità. Distribuire i nodi uniformemente tra le zone di disponibilità ridurrà notevolmente l'impatto di un eventuale, ancorché improbabile, guasto di una zona. Questa operazione viene eseguita automaticamente per le cache serverless.

Sicurezza dell'infrastruttura in AWS ElastiCache

Come servizio gestito, AWS ElastiCache è protetto dalle procedure di sicurezza di rete globali di AWS descritte nella sezione Sicurezza e conformità nel [Centro di architettura AWS](#).

Utilizza le chiamate API pubblicate da AWS per accedere a ElastiCache tramite la rete. I client devono supportare Transport Layer Security (TLS) 1.2 o versioni successive. È consigliabile TLS 1.3 o versioni successive. I client devono, inoltre, supportare le suite di cifratura con PFS (Perfect Forward Secrecy), ad esempio Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Inoltre, le richieste devono essere firmate utilizzando un ID chiave di accesso e una chiave di accesso segreta associata a un principale IAM. In alternativa, è possibile utilizzare [AWS Security Token Service](#) (AWS STS) per generare le credenziali di sicurezza temporanee per sottoscrivere le richieste.

Aggiornamenti del servizio in ElastiCache

ElastiCache monitora automaticamente la tua flotta di cache, cluster e nodi per applicare gli aggiornamenti del servizio non appena sono disponibili. Gli aggiornamenti di servizio per le cache serverless vengono applicati automaticamente e in modo trasparente. Per i cluster progettati autonomamente, è necessario impostare una finestra di manutenzione predefinita in modo da poter applicare questi aggiornamenti. ElastiCache Tuttavia, in alcuni casi questo approccio potrebbe risultare troppo rigido e vincolare per i flussi aziendali.

Con gli aggiornamenti di servizio controlli quando e quali aggiornamenti vengono applicati ai cluster progettati autonomamente. È inoltre possibile monitorare lo stato di avanzamento di questi aggiornamenti nel ElastiCache cluster selezionato in tempo reale.

Gestione degli aggiornamenti del servizio

ElastiCache gli aggiornamenti dei servizi per i cluster progettati autonomamente vengono rilasciati regolarmente. Se disponi di uno o più cluster qualificati autoprogettati per tali aggiornamenti del servizio, ricevi notifiche tramite e-mail, SNS, Personal Health Dashboard (PHD) ed CloudWatch eventi Amazon quando gli aggiornamenti vengono rilasciati. Gli aggiornamenti vengono visualizzati anche nella pagina Service Updates sulla console. ElastiCache Utilizzando questa dashboard, puoi visualizzare tutti gli aggiornamenti del servizio e il relativo stato per il tuo ElastiCache parco veicoli. Gli aggiornamenti di servizio per le cache serverless vengono applicati in modo trasparente e non possono essere gestiti tramite la pagina Aggiornamenti del servizio.

Si controlla quando applicare un aggiornamento prima dell'avvio dell'aggiornamento automatico. Ti consigliamo vivamente di applicare qualsiasi aggiornamento di tipo security-update il prima possibile

per garantire che i tuoi ElastiCache cluster siano sempre up-to-date dotati delle patch di sicurezza correnti.

Le seguenti sezioni esplorano queste opzioni in dettaglio:

Argomenti

- [Applicazione degli aggiornamenti di servizio](#)
- [Verifica dell'applicazione dell'ultimo aggiornamento del servizio tramite la console AWS](#)
- [Interruzione degli aggiornamenti del servizio](#)

Applicazione degli aggiornamenti di servizio

Puoi iniziare ad applicare gli aggiornamenti di servizio al parco istanze dal momento in cui lo stato degli aggiornamenti è available. Gli aggiornamenti di servizio sono cumulativi. In altre parole, tutti gli aggiornamenti non ancora applicati sono inclusi con l'ultimo aggiornamento.

Se per un aggiornamento del servizio è abilitato l'aggiornamento automatico, puoi scegliere di non intraprendere alcuna azione quando diventa disponibile. ElastiCache pianificherà l'applicazione dell'aggiornamento durante una delle prossime finestre di manutenzione dei cluster dopo la data di inizio dell'aggiornamento automatico. Riceverai notifiche correlate per ogni fase dell'aggiornamento.

Note

Puoi applicare solo aggiornamenti di servizio con stato availableoscheduled.

Per ulteriori informazioni sulla revisione e l'applicazione di eventuali aggiornamenti specifici del servizio ai cluster applicabili ElastiCache , consulta. [Applicazione degli aggiornamenti del servizio tramite la console](#)

Quando è disponibile un nuovo aggiornamento del servizio per uno o più ElastiCache cluster, puoi utilizzare la ElastiCache console, l'API o AWS CLI applicare l'aggiornamento. Le sezioni seguenti illustrano le opzioni che puoi utilizzare per applicare gli aggiornamenti.

Applicazione degli aggiornamenti del servizio tramite la console

Per visualizzare l'elenco degli aggiornamenti di servizio disponibili, assieme ad altre informazioni, accedi alla pagina Aggiornamenti di servizio nella console.

1. Accedi AWS Management Console e apri la ElastiCache console Amazon all'[indirizzo https://console.aws.amazon.com/elasticache/](https://console.aws.amazon.com/elasticache/).
2. Nel riquadro di navigazione, sceglie Aggiornamenti di servizio.
3. Sotto Aggiornamenti di servizio puoi visualizzare quanto segue:
 - Nome aggiornamento di servizio: il nome univoco dell'aggiornamento di servizio
 - Tipo di aggiornamento: il tipo di aggiornamento di servizio, cioè security-update o engine-update
 - Gravità aggiornamento): la priorità di applicazione dell'aggiornamento:
 - critical (critica): si consiglia di applicare immediatamente questo aggiornamento (entro 14 giorni o meno).
 - important (importante): si consiglia di applicare questo aggiornamento non appena l'attività aziendale lo consente (entro 30 giorni o meno).
 - medium (media): si consiglia di applicare questo aggiornamento il prima possibile (entro 60 giorni o meno).
 - low (bassa): si consiglia di applicare questo aggiornamento il prima possibile (entro 90 giorni o meno).
 - Versione del motore: se il tipo di aggiornamento è aggiornamento del motore, è la versione del motore che viene aggiornata.
 - Data di rilascio: indica quando l'aggiornamento è stato rilasciato per essere applicato sui cluster.
 - Data limite consigliata: data ElastiCache indicativa entro la quale applicare gli aggiornamenti.
 - Status (Stato): lo stato dell'aggiornamento, che è uno dei seguenti:
 - disponibile: l'aggiornamento è disponibile per i cluster necessari.
 - complete: l'aggiornamento è stato applicato.
 - cancelled (annullato): l'aggiornamento è stato annullato e non è più necessario.
 - expired (scaduto): l'aggiornamento non è più disponibile per l'applicazione.
4. Scegli un aggiornamento singolo (non il pulsante a sinistra) per visualizzare i dettagli dell'aggiornamento di servizio.

Nella sezione Stato aggiornamento cluster puoi visualizzare un elenco di cluster in cui l'aggiornamento di servizio non è stato applicato o è stato applicato recentemente. Per ogni cluster puoi visualizzare:

- Nome cluster: il nome del cluster
- Nodi aggiornati: il rapporto tra i singoli nodi in un cluster specifico che sono stati aggiornati o rimangono disponibili per l'aggiornamento di servizio specifico.
- Tipo di aggiornamento: il tipo di aggiornamento di servizio, cioè security-update o engine-update
- Stato: lo stato dell'aggiornamento di servizio sul cluster, cioè:
 - available: l'aggiornamento è disponibile per i cluster richiesti.
 - in corso: l'aggiornamento del cluster è in fase di applicazione.
 - scheduled (pianificato): la data di aggiornamento è stata pianificata.
 - complete (completo): l'aggiornamento è stato applicato correttamente. Il cluster con uno stato completo verrà visualizzato per 7 giorni dopo il completamento.

Se hai scelto uno o tutti i cluster con stato available o scheduled e hai selezionato Applica ora, l'aggiornamento inizierà ad essere applicato su tali cluster.

Applicazione degli aggiornamenti del servizio tramite AWS CLI

Dopo aver ricevuto la notifica che gli aggiornamenti del servizio sono disponibili, puoi esaminarli e applicarli utilizzando la AWS CLI:

- Per recuperare una descrizione degli aggiornamenti del servizio disponibili, esegui il comando seguente:

```
aws elasticache describe-service-updates --service-update-status available
```

Per ulteriori informazioni, vedere [describe-service-updates](#).

- Per applicare un aggiornamento di servizio a un elenco di cluster, utilizza il comando seguente:

```
aws elasticache batch-apply-update-action --service-update ServiceUpdateNameToApply=sample-service-update --cluster-names cluster-1 cluster2
```

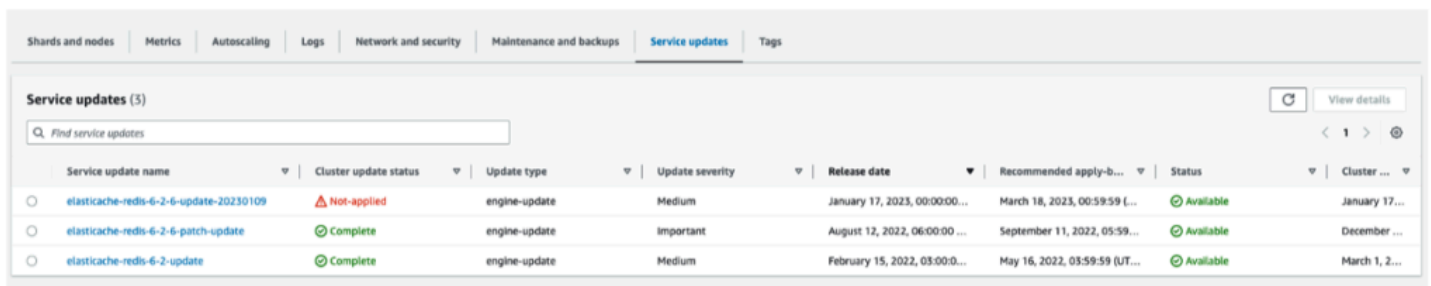
Per ulteriori informazioni, vedere [batch-apply-update-action](#).

Verifica dell'applicazione dell'ultimo aggiornamento del servizio tramite la console AWS

Puoi verificare che ElastiCache i tuoi cluster Redis stiano eseguendo l'ultimo aggiornamento del servizio seguendo questi passaggi:

1. Scegli un cluster applicabile nella pagina Redis Clusters
2. Scegli Aggiornamenti del servizio nel riquadro di navigazione per visualizzare gli eventuali aggiornamenti del servizio applicabili per quel cluster.

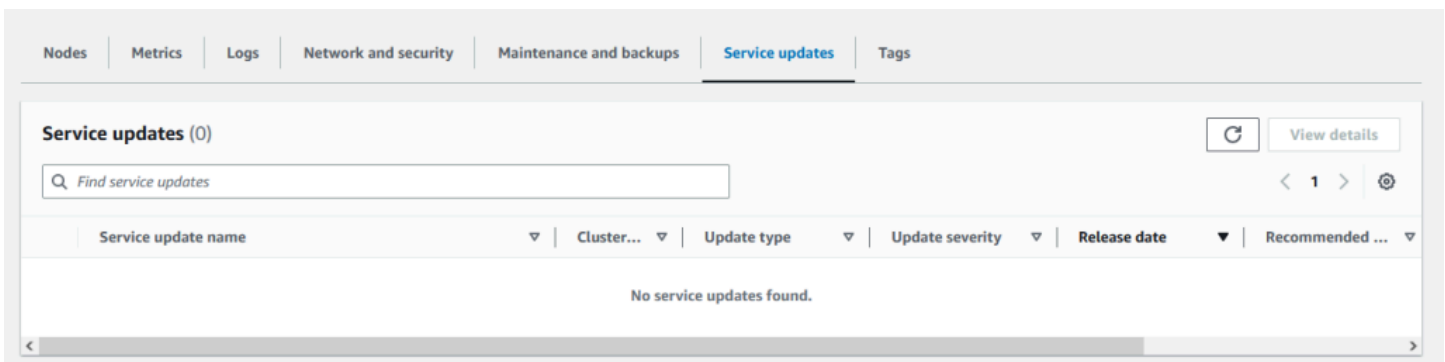
Se la console visualizza un elenco di aggiornamenti del servizio, puoi selezionare l'aggiornamento del servizio e scegliere **Applica ora**.



The screenshot shows the 'Service updates' section in the AWS console. It features a search bar and a table with the following columns: Service update name, Cluster update status, Update type, Update severity, Release date, Recommended apply-by, Status, and Cluster. Three updates are listed:

Service update name	Cluster update status	Update type	Update severity	Release date	Recommended apply-by	Status	Cluster
elasticache-redis-6-2-6-update-20230109	Not-applied	engine-update	Medium	January 17, 2023, 00:00:00...	March 18, 2023, 00:59:59 (...)	Available	January 17...
elasticache-redis-6-2-6-patch-update	Complete	engine-update	Important	August 12, 2022, 06:00:00 ...	September 11, 2022, 05:59...	Available	December ...
elasticache-redis-6-2-update	Complete	engine-update	Medium	February 15, 2022, 03:00:0...	May 16, 2022, 03:59:59 (U.T...	Available	March 1, 2...

Se sulla console viene visualizzato il messaggio «Nessun aggiornamento del servizio trovato», significa che al cluster ElastiCache for Redis è già applicato l'ultimo aggiornamento del servizio.



The screenshot shows the 'Service updates' section in the AWS console with the message 'No service updates found.' displayed in the table area. The table headers are: Service update name, Cluster..., Update type, Update severity, Release date, and Recommended ...

Interruzione degli aggiornamenti del servizio

Se necessario, puoi interrompere gli aggiornamenti ai cluster. Ad esempio, puoi interrompere gli aggiornamenti se si verifica un aumento imprevisto sui cluster che sono in fase di aggiornamento. Oppure potresti voler interrompere gli aggiornamenti se stanno impiegando troppo tempo e stanno interrompendo l'attività aziendale in un momento di picco.

L'operazione [Stopping \(Interruzione in corso\)](#) interrompe immediatamente tutti gli aggiornamenti a quei cluster e a tutti i nodi che devono ancora essere aggiornati. Tale operazione continua a completare tutti i nodi che hanno uno stato in progress (in corso). Tuttavia, cessa gli aggiornamenti ad altri nodi nello stesso cluster che hanno uno stato update available (aggiornamento disponibile) e li riporta a uno stato Stopping (Interruzione in corso).

Quando il flusso di lavoro Stopping (Interruzione in corso) è completo i nodi hanno uno stato che passa da Stopping (Interruzione in corso) allo stato Stopped (Interrotto). A seconda del flusso di lavoro dell'aggiornamento, alcuni cluster non avranno alcun nodo aggiornato. Altri cluster potrebbero includere alcuni nodi che vengono aggiornati e altri che presentano ancora uno stato update available (aggiornamento disponibile).

Puoi riprendere successivamente per completare il processo di aggiornamento come consentito dalle attività aziendali. In questo caso, seleziona i cluster applicabili su cui desidera completare gli aggiornamenti, quindi scegliere Apply Now (Applica ora). Per ulteriori informazioni, consulta [Applicazione degli aggiornamenti di servizio](#).

Utilizzo della console

È possibile interrompere un aggiornamento del servizio utilizzando la ElastiCache console. Di seguito viene illustrato come eseguire questa operazione:

- Dopo l'avanzamento di un aggiornamento del servizio su un cluster selezionato, la ElastiCache console visualizza la scheda Visualizza/Stop Update nella parte superiore della dashboard. ElastiCache
- Per interrompere l'aggiornamento, seleziona Stop Update (Interrompi aggiornamento).
- Quando interrompi l'aggiornamento, seleziona il cluster ed esamina lo stato. Lo stato viene modificato in Stopping (Interruzione in corso) ed eventualmente nello stato Stopped (Interrotto).

Utilizzando il AWS CLI

Puoi interrompere un aggiornamento del servizio utilizzando la AWS CLI. L'esempio di codice seguente mostra come eseguire tale operazione.

Per un gruppo di replica, procedi come descritto di seguito:

```
aws elasticache batch-stop-update-action --service-update-name sample-service-update --replication-group-ids my-replication-group-1 my-replication-group-2
```

Per un cluster di cache, procedi come descritto di seguito:

```
aws elasticache batch-stop-update-action --service-update-name sample-  
service-update --cache-cluster-ids my-cache-cluster-1 my-cache-cluster-2
```

Per ulteriori informazioni, vedere [BatchStopUpdateAction](#).

Registrazione di log e monitoraggio in Amazon ElastiCache

Ai fini della gestione è importante conoscere le prestazioni delle cache. ElastiCache genera le metriche pubblicate in File di log Amazon CloudWatch per il monitoraggio delle prestazioni delle cache. Inoltre, ElastiCache genera eventi quando si verificano modifiche significative sulle risorse di cache (ad esempio, viene creata una nuova cache o viene eliminata una cache).

Argomenti

- [Metriche ed eventi serverless](#)
- [Metriche ed eventi di cluster progettati autonomamente](#)
- [Registrazione delle chiamate all'API Amazon ElastiCache con AWS CloudTrail](#)
- [Registrazione delle chiamate all'API Amazon ElastiCache con AWS CloudTrail](#)

Metriche ed eventi serverless

Questa sezione descrive le metriche e gli eventi che puoi monitorare quando usi le cache serverless.

Argomenti

- [Metriche delle cache serverless](#)
- [Eventi di cache serverless](#)

Metriche delle cache serverless

Lo spazio dei nomi AWS/ElastiCache include le seguenti metriche CloudWatch per le cache serverless Memcached.

Parametro	Descrizione	Unità
BytesUsedForCache	Numero totale di byte utilizzati dai dati archiviati nella cache.	Byte
ElastiCacheProcessingUnits	Numero totale di ElastiCacheProcessingUnits (ECPU)	Conteggio

Parametro	Descrizione	Unità
	usate dalle richieste eseguite nella cache.	
SuccessfulReadRequestLatency	Latenza delle richieste di lettura completate.	Microsecondi
SuccessfulWriteRequestLatency	Latenza delle richieste di scrittura completate.	Microsecondi
TotalCmdsCount	Numero totale dei comandi eseguiti nella cache.	Conteggio
CurrConnections	Numero di connessioni client alla cache.	Conteggio
ThrottledCmds	Numero di richieste sottoposte e da ElastiCache alla limitazione (della larghezza di banda della rete) perché il carico di lavoro veniva dimensionato più velocemente di quanto ElastiCache potesse fare.	Conteggio
NewConnections	Il numero totale di connessioni accettate dal server durante questo periodo.	Conteggio
CurrItems	Il numero di elementi nella cache.	Conteggio
NetworkBytesIn	Byte totali trasferiti nella cache.	Byte
NetworkBytesOut	Byte totali trasferiti dalla cache.	Byte

Parametro	Descrizione	Unità
Espulsioni	Numero di chiavi eliminate dalla cache.	Conteggio
Reclaimed	Numero di chiavi scadute della cache.	Conteggio

Metriche a livello di comando

ElastiCache genera anche le seguenti metriche a livello di comando Memcached:

Parametro	Descrizione	Unità
CmdGet	Il numero di comandi get che la cache ha ricevuto.	Conteggio
CmdSet	Il numero di comandi stabiliti che la cache ha ricevuto.	Conteggio
CmdTouch	Il numero di comandi touch ricevuti dalla cache.	Conteggio
GetHits	Il numero di richieste di recupero che la cache ha ricevuto dove la chiave richiesta è stata trovata.	Conteggio
GetMisses	Il numero di richieste di recupero che la cache ha ricevuto dove la chiave richiesta non è stata trovata.	Conteggio
IncrHits	Il numero di richieste di incremento che la cache ha ricevuto dove la chiave richiesta è stata trovata.	Conteggio

Parametro	Descrizione	Unità
IncrMisses	Il numero di richieste di incremento che la cache ha ricevuto dove la chiave richiesta non è stata trovata.	Conteggio
DecrHits	Il numero di richieste di decremento che la cache ha ricevuto dove la chiave richiesta è stata trovata.	Conteggio
DecrMisses	Il numero di richieste di decremento che la cache ha ricevuto dove la chiave richiesta non è stata trovata.	Conteggio
DeleteHits	Il numero di richieste di eliminazione che la cache ha ricevuto dove la chiave richiesta è stata trovata.	Conteggio
DeleteMisses	Il numero di richieste di eliminazione che la cache ha ricevuto dove la chiave richiesta non è stata trovata.	Conteggio
TouchHits	Il numero di chiavi che sono state toccate e a cui è stata assegnata una nuova ora di scadenza.	Conteggio
TouchMisses	Il numero di chiavi toccate, ma non trovate.	Conteggio

Parametro	Descrizione	Unità
CasHits	Il numero di richieste Cas ricevute dalla cache dove la chiave richiesta è stata trovata e il valore Cas corrispondeva.	Conteggio
CasMisses	Il numero di richieste Cas ricevute dalla cache dove la chiave richiesta non è stata trovata.	Conteggio
CasBadval	Il numero di richieste Cas ricevute dalla cache dove il valore Cas non corrispondeva al valore Cas archiviato.	Conteggio
CmdFlush	Il numero di comandi flush che la cache ha ricevuto.	Conteggio

Eventi di cache serverless

ElastiCache registra i log eventi correlati alla cache serverless. Queste informazioni includono la data, l'ora, il nome e tipo di fonte e una descrizione dell'evento. Gli eventi possono essere facilmente recuperati dal log utilizzando la console ElastiCache, il comando `describe-events` della AWS CLI o l'operazione `DescribeEvents` dell'API ElastiCache.

Puoi scegliere di monitorare, inserire, trasformare e agire sugli eventi ElastiCache utilizzando Amazon EventBridge. Ulteriori informazioni sono disponibili nella Guida per l'utente di Amazon EventBridge all'indirizzo <https://docs.aws.amazon.com/eventbridge/latest/userguide/>.

Visualizzazione degli eventi ElastiCache (console)

Per visualizzare gli eventi utilizzando la console ElastiCache:

1. Accedi alla AWS Management Console e apri la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Per l'elenco di tutti gli eventi disponibili, scegliere Events (Eventi) nel pannello di navigazione.

3. Nella schermata Eventi, ogni riga dell'elenco rappresenta un singolo evento e indica l'origine, il tipo, l'ora GMT e la descrizione dell'evento. La voce Filter (Filtra) consente di specificare se si preferisce visualizzare in elenco tutti gli eventi o solo quelli di un tipo specifico.

Visualizzazione degli eventi ElastiCache (AWS CLI)

Per generare un elenco di eventi ElastiCache con la AWS CLI, utilizza il comando `describe-events`. Tramite parametri facoltativi è anche possibile specificare il tipo, l'intervallo di tempo, il numero massimo e altre peculiarità degli eventi da includere nell'elenco.

Il codice seguente elenca fino a 40 eventi di cache serverless.

```
aws elasticache describe-events --source-type serverless-cache --max-items 40
```

Il codice seguente elenca tutti gli eventi di cache serverless delle ultime 24 ore (1440 minuti).

```
aws elasticache describe-events --source-type serverless-cache --duration 1440
```

Eventi serverless

Questa sezione documenta i diversi tipi di eventi che potresti ricevere per le cache serverless.

Eventi di creazione di cache serverless

Tipo di evento	Descrizione	Unità	Origine	Messaggio
Creazione della cache completata	ARN della cache	creazione	Cache serverless	La cache <nome-cache> è stata creata ed è pronta per l'uso.
Creazione della cache non riuscita	ARN della cache	errore	Cache serverless	Impossibile creare la cache <nome-cache>. Gli indirizzi IP liberi sono insufficienti per creare un endpoint VPC.

Tipo di evento	Descrizione	Unità	Origine	Messaggio
Creazione della cache non riuscita	ARN della cache	errore	Cache serverless	Impossibile creare la cache <nome-cache>. Nella richiesta sono state fornite sottoreti non valide.
Creazione della cache non riuscita	ARN della cache	errore	Cache serverless	Impossibile creare la cache <nome-cache>. Il limite di quota è stato raggiunto per la creazione di un endpoint VPC.
Creazione della cache non riuscita	ARN della cache	errore	Cache serverless	Impossibile creare la cache <nome-cache>. Non si dispone delle autorizzazioni per creare un endpoint VPC.

Eventi di aggiornamento di cache serverless

Tipo di evento	Elenco delle risorse	Categoria	Origine	Messaggio
Aggiornamento della cache completato	ARN della cache	modifica della configurazione	Cache serverless	SecurityGroups aggiornati per la

Tipo di evento	Elenco delle risorse	Categoria	Origine	Messaggio
				cache <nome-cache>.
Aggiornamento della cache completato	ARN della cache	modifica della configurazione	Cache serverless	Tag aggiornati per la cache <nome-cache>.
Aggiornamento della cache non riuscito	ARN della cache	modifica della configurazione	Cache serverless	L'aggiornamento della cache <nome-cache> non è riuscito. L'aggiornamento di SecurityGroups non è riuscito.
Aggiornamento della cache non riuscito	ARN della cache	modifica della configurazione	Cache serverless	L'aggiornamento della cache <nome-cache> non è riuscito. L'aggiornamento di SecurityGroups non è riuscito a causa di autorizzazioni insufficienti.

Tipo di evento	Elenco delle risorse	Categoria	Origine	Messaggio
Aggiornamento della cache non riuscito	ARN della cache	modifica della configurazione	Cache serverless	L'aggiornamento della cache <nome-cache> non è riuscito. L'aggiornamento di SecurityGroups non è riuscito perché i SecurityGroups non sono validi.

Eventi di eliminazione di cache serverless

Tipo di evento	Elenco delle risorse	Categoria	Origine	Messaggio
Eliminazione della cache completata	ARN della cache	eliminazione	Cache serverless	La cache <nome-cache> è stata eliminata.

Eventi di limite di utilizzo di cache serverless

Tipo di evento	Descrizione	Unità	Origine	Messaggio
Aggiornamento della cache completato	ARN della cache	modifica della configurazione	Cache serverless	Limiti aggiornati per la cache <nome-cache>.
Aggiornamento della cache non riuscito	ARN della cache	errore	Cache serverless	L'aggiornamento di un limite per la cache <nome-cache> non è riuscito perché

Tipo di evento	Descrizione	Unità	Origine	Messaggio
				la cache è stata eliminata.
Aggiornamento della cache non riuscito	ARN della cache	errore	Cache serverless	L'aggiornamento di un limite per la cache <nome-cache> non è riuscito a causa di una configurazione non valida.

Metriche ed eventi di cluster progettati autonomamente

In questa sezione sono descritti le metriche, gli eventi e i log previsti quando si usano i cluster progettati autonomamente.

Argomenti

- [Metriche per cluster progettati autonomamente](#)
- [Eventi per cluster progettati autonomamente](#)
- [Monitoraggio dell'utilizzo con i parametri di CloudWatch](#)
- [Monitoraggio di Amazon SNS per gli eventi ElastiCache](#)

Metriche per cluster progettati autonomamente

Quando si progettano autonomamente i cluster, ElastiCache genera le metriche a ogni livello di nodo, incluse quelle a livello di host e della cache.

Per ulteriori informazioni sulle metriche a livello di host per Memcached, consulta [Parametri a livello di host](#).

Per ulteriori informazioni sulle metriche Memcached a livello di nodo, consulta [Parametri per Memcached](#).

Eventi per cluster progettati autonomamente

ElastiCache registra i log eventi correlati alle cache progettate autonomamente. Quando usi i cluster progettati autonomamente, puoi visualizzare gli eventi del cluster nella console ElastiCache utilizzando la AWS CLI o Amazon Simple Notification Service (SNS). Gli eventi dei cluster progettati autonomamente non vengono pubblicati su Amazon EventBridge.

Le informazioni degli eventi dei cluster progettati autonomamente includono la data, l'ora, il nome e tipo di origine e una descrizione dell'evento. Gli eventi possono essere facilmente recuperati dal log utilizzando la console ElastiCache, il comando `describe-events` della AWS CLI o l'operazione `DescribeEvents` dell'API ElastiCache.

Visualizzazione degli eventi ElastiCache (Console)

La seguente procedura consente di visualizzare eventi con la console ElastiCache.

Per visualizzare eventi utilizzando la console ElastiCache

1. Accedi alla AWS Management Console e apri la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Per l'elenco di tutti gli eventi disponibili, scegliere Events (Eventi) nel pannello di navigazione.
3. Nella schermata Eventi, ogni riga dell'elenco rappresenta un singolo evento e indica l'origine, il tipo, l'ora GMT e la descrizione dell'evento. La voce Filter (Filtra) consente di specificare se si preferisce visualizzare in elenco tutti gli eventi o solo quelli di un tipo specifico.

Visualizzazione degli eventi ElastiCache (AWS CLI)

Per generare un elenco di eventi ElastiCache con la AWS CLI, utilizza il comando `describe-events`. Tramite parametri facoltativi è anche possibile specificare il tipo, l'intervallo di tempo, il numero massimo e altre peculiarità degli eventi da includere nell'elenco.

Il codice seguente elenca fino a 40 eventi di cluster progettato autonomamente.

```
aws elasticache describe-events --source-type cache-cluster --max-items 40
```

Il codice seguente elenca tutti gli eventi di cache progettate autonomamente delle ultime 24 ore (1440 minuti).

```
aws elasticache describe-events --source-type cache-cluster --duration 1440
```


Eventi di cluster progettati autonomamente


Questa sezione contiene l'elenco degli eventi che puoi ricevere per i cluster progettati autonomamente.

I seguenti eventi ElastiCache attivano notifiche Amazon SNS. Per ulteriori informazioni sui dettagli degli eventi, consulta [Visualizzazione di eventi ElastiCache C](#).

Nome evento	Messaggio	Descrizione
ElastiCache:AddCacheNodeComplete	ElastiCache:AddCacheNodeComplete : <i>cache-cluster</i>	Un nodo di cache è stato aggiunto al cluster di cache ed è pronto per l'uso.
ElastiCache:AddCacheNodeFailed due to insufficient free IP addresses	ElastiCache:AddCacheNodeFailed : <i>cluster-name</i>	Non è stato possibile aggiungere un nodo di cache per l'insufficienza di indirizzi IP disponibili.
ElastiCache:CacheClusterParametersChanged	ElastiCache:CacheClusterParametersChanged : <i>cluster-name</i>	Uno o più parametri del cluster di cache sono stati modificati.
Elasticache:cacheclusterProvisioningComplete	ElastiCache:CacheClusterProvisioningComplete <i>cluster-name-0001-005</i>	Il provisioning di un cluster di cache è completo e i nodi di cache nel cluster sono pronti all'uso.
ElastiCache:CacheClusterProvisioningFailed due to incompatible network state	ElastiCache:CacheClusterProvisioningFailed : <i>cluster-name</i>	È stato effettuato un tentativo di avvio di un nuovo cluster di cache in un cloud privato virtuale (VPC, Virtual Private Cloud) inesistente.
ElastiCache:CacheClusterScalingComplete	CacheClusterScalingComplete : <i>cluster-name</i>	Dimensionamento di un cluster di cache completato.

Nome evento	Messaggio	Descrizione
Elasticache: cacheclusterScalingFailed	ElasticaCacheclusterScalingFailed: <i>nome-cluster</i>	Non è riuscita l'operazione di dimensionamento di un cluster di cache.
Elasticache:cacheclusterSecurityGroupModified	ElastiCache:CacheClusterSecurityGroupModified : <i>cluster-name</i>	Si è verificato uno dei seguenti eventi: <ul style="list-style-type: none">• L'elenco dei gruppi di sicurezza cache autorizzati per il cluster di cache è stato modificato.• Uno o più nuovi gruppi di sicurezza EC2 sono stati ammessi nei gruppi di sicurezza cache associati al cluster di cache.• Uno o più gruppi di sicurezza EC2 sono stati rimossi dai gruppi di sicurezza cache associati al cluster di cache.

Nome evento	Messaggio	Descrizione
ElastiCache:CacheNodeReplaceStarted	ElastiCache:CacheNodeReplaceStarted : <i>cluster-name</i>	<p>ElastiCache ha rilevato che l'host che esegue un nodo di cache è danneggiato o irraggiungibile ed è iniziata la sostituzione di tale nodo.</p> <div data-bbox="1068 495 1507 810"><p> Note</p><p>La voce DNS relativa al nodo di cache in sostituzione non cambia.</p></div> <p>Nella maggior parte dei casi, non è necessario aggiornare l'elenco dei server per i client, quando si verifica questo evento. Tuttavia, alcune librerie client della cache potrebbero smettere di utilizzare il nodo di cache nonostante la sostituzione da parte di ElastiCache; in tal caso, l'applicazione dovrà aggiornare l'elenco dei server.</p>

Nome evento	Messaggio	Descrizione
ElastiCache:CacheNodeReplaceComplete	ElastiCache:CacheNodeReplaceComplete : <i>cluster-name</i>	<p>ElastiCache ha rilevato che l'host che esegue un nodo di cache è danneggiato o irraggiungibile ed effettuato la sostituzione di tale nodo.</p> <div data-bbox="1068 495 1507 810" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; background-color: #e6f2ff;"> <p> Note</p> <p>La voce DNS relativa al nodo di cache in sostituzione non cambia.</p> </div> <p>Nella maggior parte dei casi, non è necessario aggiornare l'elenco dei server per i client, quando si verifica questo evento. Tuttavia, alcune librerie client della cache potrebbero smettere di utilizzare il nodo di cache nonostante la sostituzione da parte di ElastiCache; in tal caso, l'applicazione dovrà aggiornare l'elenco dei server.</p>
ElastiCache:cacheNodesRebooted	ElastiCache:CacheNodesRebooted : <i>cluster-name</i>	<p>Uno o più nodi di cache sono stati riavviati.</p> <p>Messaggio (Memcached): "Cache node %s shutdown" Secondo messaggio: "Cache node %s restarted"</p>

Nome evento	Messaggio	Descrizione
ElastiCache: Certifica terenewalComplete (solo Redis)	ElastiCache: Certif icateRenewalComple te	Il certificato Amazon CA è stato rinnovato correttamente.
ElastiCache:CreateReplicati onGroupComplete	ElastiCache:Create ReplicationGroupCo mplete : <i>cluster-n ame</i>	Il gruppo di replica è stato creato correttamente.
ElastiCache>DeleteCacheClus terComplete	ElastiCache>Delete CacheClusterComple te : <i>cluster-name</i>	È stata completata la procedura di eliminazione di un cluster di cache con tutti i nodi associati.
ElastiCache: FailoverC omplete (solo Redis)	ElastiCache:Failov erComplete : <i>mycluster</i>	Il failover su un nodo di replica ha avuto esito positivo.
ElastiCache:ReplicationGrou pIncreaseReplicaCountFinish ed	ElastiCache:Replic ationGroupIncrease ReplicaCountFinish ed : <i>cluster-n ame-0001-005</i>	Il numero di repliche nel cluster è stato aumentato.
Elasticache:ReplicationGrou pIncreaseReplicaCount	ElastiCache:Replic ationGroupIncrease ReplicaCountStarte d : <i>cluster-n ame-0003-004</i>	Il processo di aggiunta di repliche al cluster è iniziato.
Elasticache:NoderePlacement Canceled	ElastiCache:NodeRe placementCanceled : <i>cluster-name</i>	È stata annullata la sostituzi one programmata di un nodo nel cluster.

Nome evento	Messaggio	Descrizione
ElastiCache:NodeReplacementRescheduled	ElastiCache:NodeReplacementRescheduled : <i>cluster-name</i>	È stata riprogrammata la già prevista sostituzione di un nodo del cluster in una nuova finestra riportata nella notifica. Per informazioni sulle azioni che è possibile intraprendere, consulta Replacing cache nodes for Memcached .
ElastiCache:NodeReplacementScheduled	ElastiCache:NodeReplacementScheduled : <i>cluster-name</i>	È stata programmata la sostituzione di un nodo del cluster nella finestra riportata nella notifica. Per informazioni sulle azioni che è possibile intraprendere, consulta Replacing cache nodes for Memcached .
ElastiCache:RemoveCacheNodeComplete	ElastiCache:RemoveCacheNodeComplete : <i>cluster-name</i>	Un nodo di cache è stato rimosso dal cluster di cache.
ElastiCache:ReplicationGroupScalingComplete	ElastiCache:ReplicationGroupScalingComplete : <i>cluster-name</i>	Operazione di scalabilità verticale sul gruppo di replica completata correttamente.
ElastiCache:ReplicationGroupScalingFailed	"Failed applying modification to cache node type to %s."	Operazione di scalabilità verticale sul gruppo di replica non riuscita.
ElastiCache:ServiceUpdateAvailableForNode	"Service update is available for cache node %s."	È disponibile un aggiornamento self-service per il nodo.

Nome evento	Messaggio	Descrizione
ElastiCache: SnapshotComplete (solo Redis)	ElastiCache:SnapshotComplete : <i>cluster-name</i>	È stata correttamente acquisita una snapshot di cache.
ElastiCache: SnapshotFailed (solo Redis)	SnapshotFailed : <i>cluster-name</i>	Non è stata portata a compimento l'acquisizione di una snapshot di cache. Per informazioni dettagliate sulle cause, consulta gli eventi della cache del cluster. Lo stato della snapshot, riportato in DescribeSnapshots , sarà failed.

Monitoraggio dell'utilizzo con i parametri di CloudWatch

ElastiCache offre parametri che consentono di monitorare i cluster. Puoi accedere a questi parametri mediante CloudWatch. Per informazioni su CloudWatch, consulta la [documentazione di Amazon CloudWatch](#).

ElastiCache fornisce parametri a livello di host (ad esempio, utilizzo della CPU) e parametri che sono specifici del software del motore di cache (ad esempio, cache gets e cache misses). Questi parametri vengono misurati e pubblicati per ogni nodo di cache in intervalli di 60 secondi.

Important

Valuta se impostare allarmi CloudWatch su determinati parametri chiave, in modo da ricevere una notifica se le prestazioni del cluster di cache iniziano a peggiorare. Per ulteriori informazioni sul tagging, consulta [Quali parametri è opportuno monitorare?](#) in questa guida.

Argomenti

- [Parametri a livello di host](#)
- [Parametri per Memcached](#)

- [Quali parametri è opportuno monitorare?](#)
- [Monitoraggio dei parametri del cluster con Amazon CloudWatch](#)

Parametri a livello di host

Il namespace AWS/ElastiCache include i seguenti parametri a livello di host per i singoli nodi di cache.

Vedi anche

- [Parametri per Memcached](#)

Parametro	Descrizione	Unità
CPUUtilization	La percentuale di utilizzo CPU per l'intero host.	Percentuale
CPUCreditBalance	<p>Il numero di crediti CPU ottenuti, che un'istanza ha accumulato da quando è stata lanciata o avviata. Per le T2 Standard CPUCreditBalance include anche il numero di crediti di lancio che sono stati accumulati.</p> <p>I crediti vengono accumulati nel saldo del credito dopo che sono stati ottenuti e rimossi dal saldo del credito una volta spesi. Il saldo del credito ha un limite massimo, determinato dalla dimensione dell'istanza. Una volta che il limite viene raggiunto, tutti i nuovi crediti guadagnati vengono scartati. Per le T2 Standard, i crediti di lancio non contano per il limite.</p> <p>I crediti in CPUCreditBalance sono disponibili affinché l'istanza li spenda per andare oltre l'utilizzo di base della CPU.</p>	Crediti (vCPU/minuti)

Parametro	Descrizione	Unità
	I parametri di credito CPU sono disponibili solo con una frequenza di 5 minuti.	
CPUCreditUsage	<p>Il numero di crediti CPU spesi dall'istanza per l'utilizzo della CPU. Un credito CPU equivale a un vCPU che viene eseguito al 100% dell'utilizzo per un minuto o una combinazione equivalente di vCPU, utilizzo e tempo (per esempio, un vCPU che viene eseguito al 50% dell'utilizzo per due minuti o due vCPU che vengono eseguiti al 25% dell'utilizzo per due minuti).</p> <p>I parametri di credito CPU sono disponibili solo con una frequenza di 5 minuti. Se specifichi un periodo superiore a 5 minuti, usa la statistica Sum (somma) al posto di quella Average (media).</p>	Crediti (vCPU/minuti)
FreeableMemory	La quantità di memoria libera disponibile sull'host. Se puoi liberare spazio dai report OS ciò dipende dalla RAM, dai buffer e dalla cache.	Byte
NetworkBytesIn	Il numero di byte che l'host ha letto dalla rete.	Byte
NetworkBytesOut	Il numero di byte inviati dall'istanza su tutte le interfacce di rete.	Byte
NetworkPacketsIn	Il numero di pacchetti ricevuti dall'istanza su tutte le interfacce di rete. Questo parametro identifica il volume del traffico in entrata in termini di numero di pacchetti su una singola istanza.	Conteggio

Parametro	Descrizione	Unità
NetworkPacketsOut	Il numero di pacchetti inviati dall'istanza su tutte le interfacce di rete. Questo parametro identifica il volume del traffico in uscita in termini di numero di pacchetti su una singola istanza.	Conteggio
NetworkBandwidthInAllowanceExceeded	Numero di pacchetti modellati perché la larghezza di banda aggregata in entrata ha superato il valore massimo per l'istanza.	Conteggio
NetworkConntrackAllowanceExceeded	Numero di pacchetti modellati perché il rilevamento delle connessioni ha superato il valore massimo per l'istanza e non è stato possibile stabilire nuove connessioni. Ciò può comportare la perdita di pacchetti per il traffico da o verso l'istanza.	Conteggio
NetworkBandwidthOutAllowanceExceeded	Numero di pacchetti modellati perché la larghezza di banda aggregata in uscita ha superato il valore massimo per l'istanza.	Conteggio
Network Packets Per Second Allowance Exceeded	Numero di pacchetti modellati perché i pacchetti bidirezionali al secondo hanno superato il valore massimo per l'istanza.	Conteggio
NetworkMaxBytesIn	Il burst massimo di byte ricevuti ogni minuto.	Byte
NetworkMaxBytesOut	Il burst massimo di byte trasmessi ogni minuto.	Byte
NetworkMaxPacketsIn	Il burst massimo di pacchetti ricevuti ogni minuto.	Conteggio
NetworkMaxPacketsOut	Il burst massimo di pacchetti trasmessi ogni minuto.	Conteggio
SwapUsage	La quantità di spazio di swapping utilizzato sull'host.	Byte

Parametri per Memcached

Il namespace `AWS/ElastiCache` include i parametri Memcached descritti di seguito.

Lo spazio dei ElastiCache nomi `AWS/`include le seguenti metriche derivate dal comando `Memcached stats`. Ogni parametro è calcolato a livello del nodo di cache.

Consulta anche

- [Parametri a livello di host](#)

Parametro	Descrizione	Unità
<code>BytesReadIntoMemcached</code>	Il numero di byte che il nodo di cache ha letto dalla rete.	Byte
<code>BytesUsedForCacheItems</code>	Il numero di byte utilizzati per archiviare le voci di cache.	Byte
<code>BytesWrittenOutFromMemcached</code>	Il numero di byte che il nodo di cache ha scritto sulla rete.	Byte
<code>CasBadval</code>	Il numero di richieste Cas (verificare e impostare) che la cache ha ricevuto dove il valore Cas non corrispondeva al valore Cas archiviato.	Conteggio
<code>CasHits</code>	Il numero di richieste Cas che la cache ha ricevuto dove la chiave richiesta è stata trovata e il valore Cas corrispondeva.	Conteggio
<code>CasMisses</code>	Il numero di richieste Cas che la cache ha ricevuto dove la chiave richiesta non è stata trovata.	Conteggio
<code>CmdFlush</code>	Il numero di comandi flush che la cache ha ricevuto.	Conteggio

Parametro	Descrizione	Unità
CmdGet	Il numero di comandi get che la cache ha ricevuto.	Conteggio
CmdSet	Il numero di comandi stabiliti che la cache ha ricevuto.	Conteggio
CurrConnections	<p>Il conteggio del numero di connessioni connesse alla cache in un dato momento. ElastiCache utilizza da due a tre connessioni per monitorare il cluster.</p> <p>Memcached crea inoltre un numero di connessioni interne pari al doppio dei thread utilizzati per il tipo di nodo. Il conteggio dei thread per i vari tipi di nodo può essere osservato nel gruppo di parametri applicabile <code>Nodetype Specific Parameters</code>.</p> <p>Le connessioni totali sono la somma delle connessioni client, delle connessioni per il monitoraggio e delle connessioni interne citate sopra.</p>	Conteggio
CurrItems	Il conteggio del numero di elementi attualmente memorizzati nella cache.	Conteggio
DecrHits	Il numero di richieste di decremento che la cache ha ricevuto dove la chiave richiesta è stata trovata.	Conteggio
DecrMisses	Il numero di richieste di decremento che la cache ha ricevuto dove la chiave richiesta non è stata trovata.	Conteggio
DeleteHits	Il numero di richieste di eliminazione che la cache ha ricevuto dove la chiave richiesta è stata trovata.	Conteggio

Parametro	Descrizione	Unità
DeleteMisses	Il numero di richieste di eliminazione che la cache ha ricevuto dove la chiave richiesta non è stata trovata.	Conteggio
Evictions	Il numero di elementi non scaduti che la cache ha rimosso per creare spazio per nuove scritture.	Conteggio
GetHits	Il numero di richieste di recupero che la cache ha ricevuto dove la chiave richiesta è stata trovata.	Conteggio
GetMisses	Il numero di richieste di recupero che la cache ha ricevuto dove la chiave richiesta non è stata trovata.	Conteggio
IncrHits	Il numero di richieste di incremento che la cache ha ricevuto dove la chiave richiesta è stata trovata.	Conteggio
IncrMisses	Il numero di richieste di incremento che la cache ha ricevuto dove la chiave richiesta non è stata trovata.	Conteggio
Reclaimed	Il numero di elementi scaduti che la cache ha rimosso per creare spazio per nuove scritture.	Conteggio

Per Memcached 1.4.14 vengono forniti i seguenti parametri aggiuntivi.

Parametro	Descrizione	Unità
BytesUsedForHash	Il numero di byte attualmente utilizzati dalle tabelle hash.	Byte
CmdConfigGet	Il numero cumulativo di richieste config get.	Conteggio

Parametro	Descrizione	Unità
CmdConfigSet	Il numero cumulativo di richieste config set.	Conteggio
CmdTouch	Il numero cumulativo di richieste touch.	Conteggio
CurrConfig	Il numero corrente di configurazioni memorizzate.	Conteggio
EvictedUnfetched	Il numero di elementi validi rimossi dalla cache utilizzata per ultima (LRU) che non sono stati toccati dopo essere stati impostati.	Conteggio
ExpiredUnfetched	Il numero di elementi scaduti recuperati dalla LRU che non sono stati toccati dopo essere stati impostati.	Conteggio
SlabsMoved	Il numero totale di pagine di allocazione memoria che sono state spostate.	Conteggio
TouchHits	Il numero di chiavi che sono state toccate e a cui è stata assegnata una nuova ora di scadenza.	Conteggio
TouchMisses	Il numero di elementi che sono stati toccati, ma non trovati.	Conteggio

Lo spazio dei nomi ElastiCache AWS/include le seguenti metriche calcolate a livello di cache.

Parametro	Descrizione	Unità
NewConnections	Il numero di nuove connessioni che la cache ha ricevuto. Tale numero viene ricavato dalla statistica Memcached total_connections mediante la registrazione della modifica in total_connections per un periodo di tempo. Questo sarà sempre almeno 1, a causa di una connessione riservata a ElastiCache	Conteggio

Parametro	Descrizione	Unità
NewItems	Il numero di nuovi elementi che la cache ha memorizzato. Tale numero viene ricavato dalla statistica Memcached total_items mediante la registrazione della modifica in total_items per un periodo di tempo.	Conteggio
UnusedMemory	<p>La quantità di memoria non utilizzata dai dati. Tale numero viene ricavato dalla statistica Memcached limit_maxbytes e bytes sottraendo bytes da limit_maxbytes.</p> <p>Poiché l'overhead di Memcached utilizza memoria in aggiunta a quella utilizzata dai dati, non UnusedMemory deve essere considerata la quantità di memoria disponibile per dati aggiuntivi. Potrebbero verificarsi delle rimozioni anche se hai ancora della memoria inutilizzata.</p> <p>Per ulteriori informazioni, consulta Utilizzo della memoria degli elementi Memcached.</p>	Byte

Quali parametri è opportuno monitorare?

I parametri CloudWatch seguenti offrono informazioni dettagliate riguardo alle prestazioni di ElastiCache. Nella maggior parte dei casi, ti consigliamo di impostare allarmi CloudWatch per questi parametri in modo da poter apportare delle correzioni prima che si verifichino problemi di prestazioni.

Parametri da monitorare

- [CPUUtilization](#)
- [SwapUsage](#)
- [Espulsioni](#)
- [CurrConnections](#)

CPUUtilization

Si tratta di un parametro a livello di host restituito come una percentuale. Per ulteriori informazioni, consulta [Parametri a livello di host](#).

Poiché Memcached è multi-thread, questo parametro può arrivare a 90%. Se questa soglia viene superata, dimensiona verticalmente il cluster di cache utilizzando un tipo di nodo di cache più grande o dimensiona orizzontalmente aggiungendo più nodi di cache.

SwapUsage

Si tratta di un parametro a livello di host restituito in byte. Per ulteriori informazioni, consulta [Parametri a livello di host](#).

La metrica `FreeableMemory` CloudWatch è vicina a 0 (cioè inferiore a 100 MB) o la metrica `SwapUsage` maggiore della metrica `FreeableMemory` indica che un nodo è sotto pressione di memoria. I caso contrario, ti consigliamo di incrementare il valore del parametro `ConnectionOverhead`.

Espulsioni

Si tratta di un parametro del motore di cache. Ti consigliamo di determinare la tua soglia di allarme per questo parametro in base alle esigenze dell'applicazione.

Se la soglia scelta viene superata, dimensiona verticalmente il cluster utilizzando un tipo di nodo più grande o dimensiona orizzontalmente aggiungendo più nodi di cache.

CurrConnections

Si tratta di un parametro del motore di cache. Ti consigliamo di determinare la tua soglia di allarme per questo parametro in base alle esigenze dell'applicazione.

Un numero crescente di CurrConnections potrebbe indicare un problema con l'applicazione; per risolvere questo problema è opportuno esaminare il comportamento dell'applicazione.

Monitoraggio dei parametri del cluster con Amazon CloudWatch

ElastiCache e CloudWatch sono integrati per consentire di acquisire vari parametri. Puoi monitorare questi parametri utilizzando CloudWatch.

Note

Il seguente esempio di interfaccia riga di comando richiede gli strumenti a riga di comando CloudWatch. Per ulteriori informazioni su CloudWatch e per scaricare gli strumenti per sviluppatori, consulta la pagina del prodotto CloudWatch.

Le procedure seguenti illustrano come utilizzare CloudWatch per acquisire statistiche sullo spazio di archiviazione di un cluster di cache nell'ultima ora.

Note

I valori `StartTime` ed `EndTime` degli esempi seguenti sono forniti solo a scopo illustrativo. Dovranno quindi, all'occorrenza, essere sostituiti con gli orari di inizio e fine appropriati per i nodi di cache.

Per informazioni sui limiti ElastiCache, consulta [AWS Restrizioni dei servizi](#) per ElastiCache.

Monitoraggio dei parametri di CloudWatch Cluster e Node (console)

Per visualizzare le statistiche di utilizzo della CPU relative a un cluster di cache

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Scegliere i nodi di cache di cui visualizzare i parametri.

Note

La selezione di oltre 20 nodi disabilita la visualizzazione dei parametri sulla console.

- a. Nella pagina Cache Clusters (Cluster di cache) della console di gestione di AWS, fare clic sul nome di uno o più cluster di cache.

Viene visualizzata la pagina con i dettagli del cluster di cache.

- b. Fare clic sulla scheda Nodes (Nodi) nella parte superiore della finestra.
- c. Scegliere i nodi di cache di cui visualizzare i parametri nella scheda Nodes (Nodi) della finestra dei dettagli.

Nella parte inferiore della finestra della console viene visualizzato l'elenco dei parametri CloudWatch disponibili.

- d. Fare clic sul parametro CPU Utilization (Utilizzo CPU).

Si apre la console CloudWatch, che mostra i parametri selezionati. È possibile modificare i parametri visualizzati, mediante gli elenchi a discesa di Statistic (Statistica) e Period (Periodo) e la scheda Time Range (Intervallo di tempo).

Monitoraggio delle metriche del cluster e dei nodi CloudWatch tramite la CLI

Per visualizzare le statistiche di utilizzo della CPU relative a un cluster di cache

- Per Linux, macOS o Unix:

```
aws cloudwatch get-metric-statistics \  
  --namespace AWS/ElastiCache \  
  --metric-name CPUUtilization \  
  --dimensions='[{"Name":"CacheClusterId","Value":"test"},  
{"Name":"CacheNodeId","Value":"0001"}]' \  
  --statistics=Average \  
  --start-time 2018-07-05T00:00:00 \  
  --end-time 2018-07-06T00:00:00 \  
  --period=3600
```

Per Windows:

```
aws cloudwatch get-metric-statistics ^  
  --namespace AWS/ElastiCache ^  
  --metric-name CPUUtilization ^  
  --dimensions='[{"Name":"CacheClusterId","Value":"test"},  
{"Name":"CacheNodeId","Value":"0001"}]' ^  
  --statistics=Average ^  
  --start-time 2018-07-05T00:00:00 ^  
  --end-time 2018-07-06T00:00:00 ^
```

```
--period=3600
```

Monitoraggio delle metriche del cluster e dei nodi CloudWatch tramite l'API CloudWatch

Per visualizzare le statistiche di utilizzo della CPU relative a un cluster di cache

- Chiamare l'API di CloudWatch `GetMetricStatistics` con i parametri riportati di seguito (tenendo presente che gli orari di inizio e fine sono puramente esemplificativi e dovranno, all'occorrenza, essere sostituiti con quelli appropriati):
 - `Statistics.member.1=Average`
 - `Namespace=AWS/ElastiCache`
 - `StartTime=2013-07-05T00:00:00`
 - `EndTime=2013-07-06T00:00:00`
 - `Period=60`
 - `MeasureName=CPUUtilization`
 - `Dimensions=CacheClusterId=mycachecluster,CacheNodeId=0002`

Example

```
http://monitoring.amazonaws.com/  
  ?Action=GetMetricStatistics  
  &SignatureVersion=4  
  &Version=2014-12-01  
  &StartTime=2018-07-05T00:00:00  
  &EndTime=2018-07-06T23:59:00  
  &Period=3600  
  &Statistics.member.1=Average  
  &Dimensions.member.1="CacheClusterId=mycachecluster"  
  &Dimensions.member.2="CacheNodeId=0002"  
  &Namespace=&AWS;/ElastiCache  
  &MeasureName=CPUUtilization  
  &Timestamp=2018-07-07T17%3A48%3A21.746Z  
  &AWS;AccessKeyId=<&AWS; Access Key ID>  
  &Signature=<Signature>
```

Monitoraggio di Amazon SNS per gli eventi ElastiCache

Quando in un cluster accadono eventi significativi, ElastiCache invia una notifica a un argomento Amazon SNS specifico. Esempi includono l'impossibilità di aggiungere un nodo, l'aggiunta di un nodo, la modifica di un gruppo di sicurezza e altro ancora. Tramite il monitoraggio degli eventi chiave, è possibile conoscere lo stato corrente dei cluster e, in base all'evento, intraprendere eventuali operazioni correttive.

Argomenti

- [Gestione ElastiCache Amazon SNS](#)
- [Visualizzazione di eventi ElastiCache C](#)
- [Notifiche di eventi Amazon SNS](#)

Gestione ElastiCache Amazon SNS

Puoi configurare ElastiCache in modo tale che invii notifiche sugli eventi importanti del cluster con Amazon Simple Notification Service (Amazon SNS). Negli esempi che seguono, verrà configurato un cluster con l'Amazon Resource Name (ARN) di un argomento Amazon SNS per la ricezione di notifiche.

Note


Tale argomento presuppone l'avvenuta registrazione a Amazon SNS, nonché la configurazione e sottoscrizione di un argomento Amazon SNS. Per ulteriori informazioni su come procedere, consultare la [Guida per gli sviluppatori di Amazon Simple Notification Service](#).

Aggiungere un argomento Amazon SNS.

Le sezioni che seguono illustrano come aggiungere un argomento Amazon SNS con la console AWS, la AWS CLI o l'API di ElastiCache.

Aggiunta di un argomento Amazon SNS (console)

Nella seguente procedura viene mostrato come aggiungere un argomento Amazon SNS a un cluster.

 Note

Attenendosi alla presente procedura, è anche possibile modificare l'argomento Amazon SNS.

Per aggiungere o modificare l'argomento Amazon SNS per un cluster (console)

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. In Clusters (Cluster), scegliere il cluster al quale aggiungere o di cui modificare un ARN d'argomento Amazon SNS.
3. Scegli Modifica.
4. In Modify Cluster (Modifica cluster) nella sezione Topic for SNS Notification (Argomento per notifica SNS), scegliere l'argomento SNS da aggiungere o scegliere Manual ARN input (Input ARN manuale) e digitare l'ARN dell'argomento Amazon SNS.
5. Scegli Modifica.

Aggiunta di un argomento Amazon SNS (AWS CLI)

Per aggiungere o modificare l'argomento Amazon SNS di un cluster, utilizzare il comando AWS CLI del `modify-cache-cluster`.

L'esempio di codice riportato di seguito rappresenta l'aggiunta di un ARN d'argomento Amazon SNS a `my-cluster`.

Per Linux, macOS o Unix:

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-cluster \  
  --notification-topic-arn arn:aws:sns:us-west-2:123456789xxx:ElastiCacheNotifications
```

Per Windows:

```
aws elasticache modify-cache-cluster ^  
  --cache-cluster-id my-cluster ^  
  --notification-topic-arn arn:aws:sns:us-west-2:123456789xx:ElastiCacheNotifications
```

Per ulteriori informazioni, consulta [modify-cache-cluster](#).

Aggiunta di un argomento Amazon SNS (API ElastiCache)

Per aggiungere o modificare l'argomento Amazon SNS di un cluster, occorre chiamare l'operazione `ModifyCacheCluster` con i seguenti parametri:

- `CacheClusterId=my-cluster`
- `TopicArn=arn%3Aaws%3Asns%3Aus-west-2%3A565419523791%3AElastiCacheNotifications`

Example

```
https://elasticache.amazonaws.com/  
  ?Action=ModifyCacheCluster  
  &ApplyImmediately=false  
  &CacheClusterId=my-cluster  
  &NotificationTopicArn=arn%3Aaws%3Asns%3Aus-west-2%3A565419523791%3AElastiCacheNotifications  
  &Version=2014-12-01  
  &SignatureVersion=4  
  &SignatureMethod=HmacSHA256  
  &Timestamp=20141201T220302Z  
  &X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
  &X-Amz-Date=20141201T220302Z  
  &X-Amz-SignedHeaders=Host  
  &X-Amz-Expires=20141201T220302Z  
  &X-Amz-Credential=<credential>  
  &X-Amz-Signature=<signature>
```

Per ulteriori informazioni, consulta [ModifyCacheCluster](#).

Attivazione e disattivazione delle notifiche Amazon SNS

È possibile, in base alle proprie esigenze, abilitare o disabilitare le notifiche relative a un cluster. La seguente procedura mostra come disabilitare le notifiche Amazon SNS.

Attivazione e disattivazione delle notifiche Amazon SNS (Console)

Per disabilitare le notifiche di Amazon SNS con la AWS Management Console

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.

2. Per visualizzare l'elenco dei cluster che eseguono Memcached, scegliere Memcached (Memcached) nel pannello di navigazione.
3. Scegliere la casella a sinistra del cluster di cui modificare le notifiche.
4. Scegli Modifica.
5. In Modify Cluster (Modifica cluster) nella sezione Topic for SNS Notification (Argomento per notifica SNS), scegliere Disable Notifications (Disabilita notifiche).
6. Scegli Modifica.

Attivazione e disattivazione delle notifiche Amazon SNS (AWS CLI)

Per disabilitare le notifiche Amazon SNS, occorre utilizzare il comando `modify-cache-cluster` con i seguenti parametri:

Per Linux, macOS o Unix:

```
aws elasticache modify-cache-cluster \  
  --cache-cluster-id my-cluster \  
  --notification-topic-status inactive
```

Per Windows:

```
aws elasticache modify-cache-cluster ^  
  --cache-cluster-id my-cluster ^  
  --notification-topic-status inactive
```

Attivazione e disattivazione delle notifiche Amazon SNS (API ElastiCache)

Per disabilitare le notifiche Amazon SNS, occorre chiamare l'operazione `ModifyCacheCluster` con i seguenti parametri:

- `CacheClusterId=my-cluster`
- `NotificationTopicStatus=inactive`

Questa chiamata restituisce un output simile al seguente:

Example

```
https://elasticache.us-west-2.amazonaws.com/
```

```
?Action=ModifyCacheCluster
&ApplyImmediately=false
&CacheClusterId=my-cluster
&NotificationTopicStatus=inactive
&Version=2014-12-01
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&Timestamp=20141201T220302Z
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256
&X-Amz-Date=20141201T220302Z
&X-Amz-SignedHeaders=Host
&X-Amz-Expires=20141201T220302Z
&X-Amz-Credential=<credential>
&X-Amz-Signature=<signature>
```

Visualizzazione di eventi ElastiCache C

ElastiCache registra gli eventi correlati alle istanze, nonché ai gruppi di sicurezza e di parametri del cluster. Queste informazioni includono la data, l'ora, il nome e tipo di fonte e una descrizione dell'evento. Gli eventi possono essere facilmente recuperati dal registro utilizzando la console di ElastiCache, il comando `describe-events` di AWS CLI o l'operazione API `DescribeEvents` di ElastiCache.

Le procedure seguenti mostrano come visualizzare tutti gli eventi ElastiCache delle ultime 24 ore (1440 minuti).

Visualizzazione degli eventi ElastiCache (Console)

La seguente procedura consente di visualizzare eventi con la console ElastiCache.

Per visualizzare eventi utilizzando la console ElastiCache

1. Accedere alla AWS Management Console e aprire la console ElastiCache all'indirizzo <https://console.aws.amazon.com/elasticache/>.
2. Per l'elenco di tutti gli eventi disponibili, scegliere Events (Eventi) nel pannello di navigazione.

Nella schermata Eventi, ogni riga dell'elenco rappresenta un evento e di quest'ultimo riporta l'fonte, il tipo (`cache-cluster`, `cache-parameter-group`, `cache-security-group` o `cache-subnet-group`), l'orario nel fuso GMT e la descrizione.

La voce Filter (Filtra) consente di specificare se si preferisce visualizzare in elenco tutti gli eventi o solo quelli di un tipo specifico.

Visualizzazione degli eventi ElastiCache (AWS CLI)

Per generare una lista di eventi ElastiCache usando la AWS CLI , utilizzare il comando `describe-events`. Tramite parametri facoltativi è anche possibile specificare il tipo, l'intervallo di tempo, il numero massimo e altre peculiarità degli eventi da includere nell'elenco.

Il codice seguente elenca fino a 40 eventi del cluster di cache.

```
aws elasticache describe-events --source-type cache-cluster --max-items 40
```

Il codice seguente elenca tutti gli eventi delle ultime 24 ore (1440 minuti).

```
aws elasticache describe-events --source-type cache-cluster --duration 1440
```

L'output del comando `describe-events` è simile a quello riportato.

```
aws elasticache describe-events --source-type cache-cluster --max-items 40
{
  "Events": [
    {
      "SourceIdentifier": "my-mem-cluster",
      "SourceType": "cache-cluster",
      "Message": "Finished modifying number of nodes from 1 to 3",
      "Date": "2020-06-09T02:01:21.772Z"
    },
    {
      "SourceIdentifier": "my-mem-cluster",
      "SourceType": "cache-cluster",
      "Message": "Added cache node 0002 in availability zone us-west-2a",
      "Date": "2020-06-09T02:01:21.716Z"
    },
    {
      "SourceIdentifier": "my-mem-cluster",
      "SourceType": "cache-cluster",
      "Message": "Added cache node 0003 in availability zone us-west-2a",
      "Date": "2020-06-09T02:01:21.706Z"
    },
    {
      "SourceIdentifier": "my-mem-cluster",
      "SourceType": "cache-cluster",
      "Message": "Increasing number of requested nodes",
      "Date": "2020-06-09T01:58:34.178Z"
    },
    {
      "SourceIdentifier": "mycluster-0003-004",
      "SourceType": "cache-cluster",
      "Message": "Added cache node 0001 in availability zone us-west-2c",
      "Date": "2020-06-09T01:51:14.120Z"
    },
    {
      "SourceIdentifier": "mycluster-0003-004",
      "SourceType": "cache-cluster",
      "Message": "This cache cluster does not support persistence (ex:
      'appendonly'). Please use a different instance type to enable persistence.",
      "Date": "2020-06-09T01:51:14.095Z"
    }
  ]
}
```

```
  },
  {
    "SourceIdentifier": "mycluster-0003-004",
    "SourceType": "cache-cluster",
    "Message": "Cache cluster created",
    "Date": "2020-06-09T01:51:14.094Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-005",
    "SourceType": "cache-cluster",
    "Message": "Added cache node 0001 in availability zone us-west-2b",
    "Date": "2020-06-09T01:42:55.603Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-005",
    "SourceType": "cache-cluster",
    "Message": "This cache cluster does not support persistence (ex:
'appendonly'). Please use a different instance type to enable persistence.",
    "Date": "2020-06-09T01:42:55.576Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-005",
    "SourceType": "cache-cluster",
    "Message": "Cache cluster created",
    "Date": "2020-06-09T01:42:55.574Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-004",
    "SourceType": "cache-cluster",
    "Message": "Added cache node 0001 in availability zone us-west-2b",
    "Date": "2020-06-09T01:28:40.798Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-004",
    "SourceType": "cache-cluster",
    "Message": "This cache cluster does not support persistence (ex:
'appendonly'). Please use a different instance type to enable persistence.",
    "Date": "2020-06-09T01:28:40.775Z"
  },
  {
    "SourceIdentifier": "mycluster-0001-004",
    "SourceType": "cache-cluster",
    "Message": "Cache cluster created",
    "Date": "2020-06-09T01:28:40.773Z"
  }
}
```

```

    }
  ]
}

```

Per ulteriori informazioni, tra cui i parametri disponibili e i valori consentiti per tali parametri, consulta [describe-events](#).

Visualizzazione degli eventi ElastiCache (API ElastiCache)

Per generare una lista di eventi ElastiCache usando l' API di ElastiCache , utilizzare l'operazione DescribeEvents. Tramite parametri facoltativi è anche possibile specificare il tipo, l'intervallo di tempo, il numero massimo e altre peculiarità degli eventi da includere nell'elenco.

Il codice seguente elenca i 40 eventi cache-cluster più recenti.

```

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeEvents
&MaxRecords=40
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&SourceType=cache-cluster
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>

```

Il codice seguente elenca gli eventi cache-cluster delle ultime 24 ore (1440 minuti).

```

https://elasticache.us-west-2.amazonaws.com/
?Action=DescribeEvents
&Duration=1440
&SignatureVersion=4
&SignatureMethod=HmacSHA256
&SourceType=cache-cluster
&Timestamp=20150202T192317Z
&Version=2015-02-02
&X-Amz-Credential=<credential>

```

Le operazioni descritte in precedenza dovrebbero generare un output simile al seguente.

```

<DescribeEventsResponse xmlns="http://elasticache.amazonaws.com/doc/2015-02-02/">
  <DescribeEventsResult>

```



```
<Events>
  <Event>
    <Message>Cache cluster created</Message>
    <SourceType>cache-cluster</SourceType>
    <Date>2015-02-02T18:22:18.202Z</Date>
    <SourceIdentifier>mem01</SourceIdentifier>
  </Event>

(...output omitted...)

</Events>
</DescribeEventsResult>
<ResponseMetadata>
  <RequestId>e21c81b4-b9cd-11e3-8a16-7978bb24ffdf</RequestId>
</ResponseMetadata>
</DescribeEventsResponse>
```

Per ulteriori informazioni, tra cui i parametri disponibili e i valori consentiti per tali parametri, consulta [DescribeEvents](#).

Notifiche di eventi Amazon SNS

Quando si verificano eventi significativi in un cluster di cache, ElastiCache può inviare messaggi tramite Amazon Simple Notification Service (SNS). Questa caratteristica può essere utilizzata per aggiornare gli elenchi dei server sulle macchine client connesse a endpoint di singoli nodi di cache afferenti a un cluster di cache.

Note

Per ulteriori informazioni su Amazon Simple Notification Service (SNS) e relativi prezzi e per i link alla documentazione Amazon SNS, consulta la [Pagina del prodotto Amazon SNS](#).

Le notifiche vengono pubblicate su un Amazon SNS specificato `Argomento`. Di seguito sono riportati i requisiti delle notifiche:

- È possibile configurare un solo argomento per le notifiche ElastiCache.
- L'account AWS titolare dell'argomento Amazon SNS deve corrispondere al proprietario del cluster di cache su cui sono abilitate le notifiche.
- L'argomento Amazon SNS che si sta pubblicando non può essere crittografato.

Note

È possibile allegare un argomento di Amazon SNS crittografato (a riposo) al cluster. Tuttavia, lo stato dell'argomento dalla console ElastiCache verrà visualizzato come inattivo, il che disassocia efficacemente l'argomento dal cluster quando ElastiCache invia i messaggi all'argomento.


- L'argomento Amazon SNS deve trovarsi nella stessa regione del cluster ElastiCache.


ElastiCache

I seguenti eventi ElastiCache attivano notifiche Amazon SNS. Per ulteriori informazioni sui dettagli degli eventi, consulta [Visualizzazione di eventi ElastiCache C](#).

Nome evento	Messaggio	Descrizione
ElastiCache:AddCacheNodeComplete	ElastiCache:AddCacheNodeComplete : <i>cache-cluster</i>	Un nodo di cache è stato aggiunto al cluster di cache ed è pronto per l'uso.
ElastiCache:AddCacheNodeFailed due to insufficient free IP addresses	ElastiCache:AddCacheNodeFailed : <i>cluster-name</i>	Non è stato possibile aggiungere un nodo di cache per l'insufficienza di indirizzi IP disponibili.
ElastiCache:CacheClusterParametersChanged	ElastiCache:CacheClusterParametersChanged : <i>cluster-name</i>	Uno o più parametri del cluster di cache sono stati modificati.
ElastiCache:cacheclusterProvisioningComplete	ElastiCache:CacheClusterProvisioningComplete <i>cluster-name-0001-005</i>	Il provisioning di un cluster di cache è completo e i nodi di cache nel cluster sono pronti all'uso.
ElastiCache:CacheClusterProvisioningFailed due to incompatible network state	ElastiCache:CacheClusterProvisioningFailed	È stato effettuato un tentativo di avvio di un nuovo cluster di cache in un cloud privato.

Nome evento	Messaggio	Descrizione
	Failed : <i>cluster-name</i>	virtuale (VPC, Virtual Private Cloud) inesistente.
ElastiCache:CacheClusterScalingComplete	CacheClusterScalingComplete : <i>cluster-name</i>	Dimensionamento di un cluster di cache completato.
Elasticache: cacheclusterScalingFailed	ElasticaCacheclusterScalingFailed: <i>nome-cluster</i>	Non è riuscita l'operazione di dimensionamento di un cluster di cache.
Elasticache:cacheclusterSecurityGroupModified	ElastiCache:CacheClusterSecurityGroupModified : <i>cluster-name</i>	<p>Si è verificato uno dei seguenti eventi:</p> <ul style="list-style-type: none"> • L'elenco dei gruppi di sicurezza cache autorizzati per il cluster di cache è stato modificato. • Uno o più nuovi gruppi di sicurezza EC2 sono stati ammessi nei gruppi di sicurezza cache associati al cluster di cache. • Uno o più gruppi di sicurezza EC2 sono stati rimossi dai gruppi di sicurezza cache associati al cluster di cache.

Nome evento	Messaggio	Descrizione
ElastiCache:CacheNodeReplaceStarted	ElastiCache:CacheNodeReplaceStarted : <i>cluster-name</i>	<p>ElastiCache ha rilevato che l'host che esegue un nodo di cache è danneggiato o irraggiungibile ed è iniziata la sostituzione di tale nodo.</p> <div data-bbox="1068 495 1507 810"><p> Note</p><p>La voce DNS relativa al nodo di cache in sostituzione non cambia.</p></div> <p>Nella maggior parte dei casi, non è necessario aggiornare l'elenco dei server per i client, quando si verifica questo evento. Tuttavia, alcune librerie client della cache potrebbero smettere di utilizzare il nodo di cache nonostante la sostituzione da parte di ElastiCache; in tal caso, l'applicazione dovrà aggiornare l'elenco dei server.</p>

Nome evento	Messaggio	Descrizione
ElastiCache:CacheNodeReplaceComplete	ElastiCache:CacheNodeReplaceComplete : <i>cluster-name</i>	<p>ElastiCache ha rilevato che l'host che esegue un nodo di cache è danneggiato o irraggiungibile ed effettuato la sostituzione di tale nodo.</p> <div data-bbox="1068 495 1507 810" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; background-color: #e6f2ff;"> <p> Note</p> <p>La voce DNS relativa al nodo di cache in sostituzione non cambia.</p> </div> <p>Nella maggior parte dei casi, non è necessario aggiornare l'elenco dei server per i client, quando si verifica questo evento. Tuttavia, alcune librerie client della cache potrebbero smettere di utilizzare il nodo di cache nonostante la sostituzione da parte di ElastiCache; in tal caso, l'applicazione dovrà aggiornare l'elenco dei server.</p>
ElastiCache:cacheNodesRebooted	ElastiCache:CacheNodesRebooted : <i>cluster-name</i>	<p>Uno o più nodi di cache sono stati riavviati.</p> <p>Messaggio (Memcached): "Cache node %s shutdown" Secondo messaggio: "Cache node %s restarted"</p>

Nome evento	Messaggio	Descrizione
ElastiCache: Certifica terenewalComplete (solo Redis)	ElastiCache: Certif icateRenewalComple te	Il certificato Amazon CA è stato rinnovato correttamente.
ElastiCache:CreateReplicati onGroupComplete	ElastiCache:Create ReplicationGroupCo mplete : <i>cluster-n ame</i>	Il gruppo di replica è stato creato correttamente.
ElastiCache>DeleteCacheClus terComplete	ElastiCache>Delete CacheClusterComple te : <i>cluster-name</i>	È stata completata la procedura di eliminazione di un cluster di cache con tutti i nodi associati.
ElastiCache: FailoverC omplete (solo Redis)	ElastiCache:Failov erComplete : <i>mycluster</i>	Il failover su un nodo di replica ha avuto esito positivo.
ElastiCache:ReplicationGrou pIncreaseReplicaCountFinish ed	ElastiCache:Replic ationGroupIncrease ReplicaCountFinish ed : <i>cluster-n ame-0001-005</i>	Il numero di repliche nel cluster è stato aumentato.
Elasticache:ReplicationGrou pIncreaseReplicaCount	ElastiCache:Replic ationGroupIncrease ReplicaCountStarte d : <i>cluster-n ame-0003-004</i>	Il processo di aggiunta di repliche al cluster è iniziato.
Elasticache:NoderePlacement Canceled	ElastiCache:NodeRe placementCanceled : <i>cluster-name</i>	È stata annullata la sostituzi one programmata di un nodo nel cluster.

Nome evento	Messaggio	Descrizione
ElastiCache:NodeReplacementRescheduled	ElastiCache:NodeReplacementRescheduled : <i>cluster-name</i>	È stata riprogrammata la già prevista sostituzione di un nodo del cluster in una nuova finestra riportata nella notifica. Per informazioni sulle azioni che è possibile intraprendere, consulta Replacing cache nodes for Memcached .
ElastiCache:NodeReplacementScheduled	ElastiCache:NodeReplacementScheduled : <i>cluster-name</i>	È stata programmata la sostituzione di un nodo del cluster nella finestra riportata nella notifica. Per informazioni sulle azioni che è possibile intraprendere, consulta Replacing cache nodes for Memcached .
ElastiCache:RemoveCacheNodeComplete	ElastiCache:RemoveCacheNodeComplete : <i>cluster-name</i>	Un nodo di cache è stato rimosso dal cluster di cache.
ElastiCache:ReplicationGroupScalingComplete	ElastiCache:ReplicationGroupScalingComplete : <i>cluster-name</i>	Operazione di scalabilità verticale sul gruppo di replica completata correttamente.
Elasticache:ReplicationGroupScalingFailed	"Failed applying modification to cache node type to %s."	Operazione di scalabilità verticale sul gruppo di replica non riuscita.
Elasticache:ServiceUpdateAvailableForNode	"Service update is available for cache node %s."	È disponibile un aggiornamento self-service per il nodo.

Nome evento	Messaggio	Descrizione
ElastiCache: SnapshotComplete (solo Redis)	ElastiCache:SnapshotComplete : <i>cluster-name</i>	È stata correttamente acquisita una snapshot di cache.
ElastiCache: SnapshotFailed (solo Redis)	SnapshotFailed : <i>cluster-name</i>	Non è stata portata a compimento l'acquisizione di una snapshot di cache. Per informazioni dettagliate sulle cause, consulta gli eventi della cache del cluster. Lo stato della snapshot, riportato in DescribeSnapshots , sarà failed.

Argomenti correlati

- [Visualizzazione di eventi ElastiCache C](#)

Registrazione delle chiamate all'API Amazon ElastiCache con AWS CloudTrail

Amazon ElastiCache è integrato con AWS CloudTrail, un servizio che offre un registro record delle operazioni eseguite da un utente, un ruolo o un servizio AWS in Amazon ElastiCache. CloudTrail acquisisce tutte le chiamate API per Amazon ElastiCache come eventi, incluse le chiamate dalla console Amazon ElastiCache e dal codice alle operazioni API di Amazon ElastiCache. Se crei un percorso, puoi abilitare la distribuzione continua di eventi CloudTrail in un bucket Amazon S3, inclusi gli eventi per Amazon ElastiCache. Se non configuri un percorso, puoi comunque visualizzare gli eventi più recenti nella console di CloudTrail in Cronologia eventi. Le informazioni raccolte da CloudTrail consentono di determinare la richiesta effettuata ad Amazon ElastiCache, l'indirizzo IP da cui è partita la richiesta, l'autore della richiesta, il momento in cui è stata eseguita e altri dettagli.

Per ulteriori informazioni su CloudTrail, consultare la [AWS CloudTrail Guida per l'utente di](#).

Informazioni su Amazon ElastiCache in CloudTrail

CloudTrail è abilitato sull'account AWS al momento della sua creazione. Quando si verifica un'attività in Amazon ElastiCache, questa viene registrata in un evento CloudTrail insieme ad altri eventi del servizio AWS nella cronologia degli eventi. È possibile visualizzare, cercare e scaricare gli eventi recenti nell'account AWS. Per ulteriori informazioni, consulta [Visualizzazione di eventi nella cronologia degli eventi di CloudTrail](#).

Per una registrazione continua degli eventi nell'account AWS, inclusi gli eventi per Amazon ElastiCache, crea un percorso. Un percorso abilita la distribuzione da parte di CloudTrail dei file di log in un bucket Amazon S3. Per impostazione di default, quando crei un trail nella console, il trail sarà valido in tutte le regioni. Il percorso registra gli eventi di tutte le Regioni nella partizione AWS e distribuisce i file di log nel bucket Amazon S3 specificato. Inoltre, è possibile configurare altri servizi AWS per analizzare con maggiore dettaglio e usare i dati evento raccolti nei log CloudTrail. Per ulteriori informazioni, consulta gli argomenti seguenti:

- [Panoramica della creazione di un percorso](#)
- [Servizi e integrazioni CloudTrail supportati](#)
- [Configurazione delle notifiche Amazon SNS per CloudTrail](#)
- [Ricezione di file di log CloudTrail da più regioni](#) e [Ricezione di file di log CloudTrail da più account](#)

Tutte le operazioni Amazon ElastiCache vengono registrate da CloudTrail e sono documentate nella [Documentazione di riferimento delle API di ElastiCache](#). Ad esempio, le chiamate alle operazioni `CreateCacheCluster`, `DescribeCacheCluster` e `ModifyCacheCluster` generano voci nei file di log di CloudTrail.

Ogni evento o voce di log contiene informazioni sull'utente che ha generato la richiesta. Le informazioni di identità consentono di determinare quanto segue:

- Se la richiesta è stata effettuata con le credenziali dell'utente IAM o root.
- Se la richiesta è stata effettuata con le credenziali di sicurezza temporanee per un ruolo o un utente federato.
- Se la richiesta è stata effettuata da un altro servizio AWS.

Per ulteriori informazioni, consulta [Elemento userIdentity di CloudTrail](#).

Informazioni sulle voci dei file di log di Amazon ElastiCache

Un percorso è una configurazione che consente la distribuzione di eventi come i file di log in un bucket Amazon S3 specificato. I file di log di CloudTrail possono contenere una o più voci di log. Un evento rappresenta una singola richiesta da un'fonte e include informazioni sul operazione richiesta, data e ora dell'operazione, parametri richiesti e così via. I file di log di CloudTrail non sono una traccia stack ordinata delle chiamate pubbliche dell'API, quindi non vengono visualizzati in un ordine specifico.

L'esempio seguente mostra una voce di log di CloudTrail che illustra l'operazione `CreateCacheCluster`.

```
{
  "eventVersion":"1.01",
  "userIdentity":{
    "type":"IAMUser",
    "principalId":"EXAMPLEEXAMPLEEXAMPLE",
    "arn":"arn:aws:iam:123456789012:user/elasticache-allow",
    "accountId":"123456789012",
    "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
    "userName":"elasticache-allow"
  },
  "eventTime":"2014-12-01T22:00:35Z",
  "eventSource":"elasticache.amazonaws.com",
  "eventName":"CreateCacheCluster",
  "awsRegion":"us-west-2",
  "sourceIPAddress":"192.0.2.01",
  "userAgent":"AWS CLI/ElastiCache 1.10 API 2014-12-01",
  "requestParameters":{
    "numCacheNodes":2,
    "cacheClusterId":"test-memcached",
    "engine":"memcached",
    "aZMode":"cross-az",
    "cacheNodeType":"cache.m1.small",
  },
  "responseElements":{
    "engine":"memcached",
    "clientDownloadLandingPage":"https://console.aws.amazon.com/elasticache/home#client-download:",
    "cacheParameterGroup":{
      "cacheParameterGroupName":"default.memcached1.4",
    }
  }
}
```

```

    "cacheNodeIdsToReboot":{
      },
    "parameterApplyStatus":"in-sync"
  },
  "preferredAvailabilityZone":"Multiple",
  "numCacheNodes":2,
  "cacheNodeType":"cache.m1.small",

  "cacheClusterStatus":"creating",
  "autoMinorVersionUpgrade":true,
  "preferredMaintenanceWindow":"thu:05:00-thu:06:00",
  "cacheClusterId":"test-memcached",
  "engineVersion":"1.4.14",
  "cacheSecurityGroups":[
    {
      "status":"active",
      "cacheSecurityGroupName":"default"
    }
  ],
  "pendingModifiedValues":{
  }
},
"requestID":"104f30b3-3548-11e4-b7b8-6d79ffe84edd",
"eventID":"92762127-7a68-42ce-8787-927d2174cde1"
}

```

L'esempio seguente mostra una voce di log di CloudTrail che illustra l'operazione `DescribeCacheCluster`. Tieni presente che per tutte le chiamate "Describe" Amazon ElastiCache (Describe*), la sezione `ResponseElements` viene rimossa e appare come `null`.

```

{
  "eventVersion":"1.01",
  "userIdentity":{
    "type":"IAMUser",
    "principalId":"EXAMPLEEXAMPLEEXAMPLE",
    "arn":"arn:aws:iam::123456789012:user/elasticache-allow",
    "accountId":"123456789012",
    "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
    "userName":"elasticache-allow"
  },
  "eventTime":"2014-12-01T22:01:00Z",
  "eventSource":"elasticache.amazonaws.com",
  "eventName":"DescribeCacheClusters",

```

```
"awsRegion":"us-west-2",
"sourceIPAddress":"192.0.2.01",
"userAgent":"AWS CLI/ElastiCache 1.10 API 2014-12-01",
"requestParameters":{"
  "showCacheNodeInfo":false,
  "maxRecords":100
},
"responseElements":null,
"requestID":"1f0b5031-3548-11e4-9376-c1d979ba565a",
"eventID":"a58572a8-e81b-4100-8e00-1797ed19d172"
}
```

L'esempio seguente mostra una voce di log di CloudTrail che registra un'operazione `ModifyCacheCluster`.

```
{
  "eventVersion":"1.01",
  "userIdentity":{"
    "type":"IAMUser",
    "principalId":"EXAMPLEEXAMPLEEXAMPLE",
    "arn":"arn:aws:iam::123456789012:user/elasticache-allow",
    "accountId":"123456789012",
    "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
    "userName":"elasticache-allow"
  },
  "eventTime":"2014-12-01T22:32:21Z",
  "eventSource":"elasticache.amazonaws.com",
  "eventName":"ModifyCacheCluster",
  "awsRegion":"us-west-2",
  "sourceIPAddress":"192.0.2.01",
  "userAgent":"AWS CLI/ElastiCache 1.10 API 2014-12-01",
  "requestParameters":{"
    "applyImmediately":true,
    "numCacheNodes":3,
    "cacheClusterId":"test-memcached"
  },
  "responseElements":{"
    "engine":"memcached",
    "clientDownloadLandingPage":"https://console.aws.amazon.com/elasticache/home#client-download:",
    "cacheParameterGroup":{"
      "cacheParameterGroupName":"default.memcached1.4",
      "cacheNodeIdsToReboot":{"
```

```
    },
    "parameterApplyStatus":"in-sync"
  },
  "cacheClusterCreateTime":"Dec 1, 2014 10:16:06 PM",
  "preferredAvailabilityZone":"Multiple",
  "numCacheNodes":2,
  "cacheNodeType":"cache.m1.small",
  "cacheClusterStatus":"modifying",
  "autoMinorVersionUpgrade":true,
  "preferredMaintenanceWindow":"thu:05:00-thu:06:00",
  "cacheClusterId":"test-memcached",
  "engineVersion":"1.4.14",
  "cacheSecurityGroups":[
    {
      "status":"active",
      "cacheSecurityGroupName":"default"
    }
  ],
  "configurationEndpoint":{
    "address":"test-memcached.example.cfg.use1prod.cache.amazonaws.com",
    "port":11211
  },
  "pendingModifiedValues":{
    "numCacheNodes":3
  }
},
"requestID":"807f4bc3-354c-11e4-9376-c1d979ba565a",
"eventID":"e9163565-376f-4223-96e9-9f50528da645"
}
```

Quote ElastiCache

L'account AWS dispone delle seguenti quote predefinite, precedentemente definite limiti, per ogni servizio AWS. Salvo dove diversamente specificato, ogni quota si applica a una regione specifica. Se per alcune quote è possibile richiedere aumenti, altre quote non possono essere modificate.

Per visualizzare le quote per ElastiCache, aprire il [Console Service Quotas](#). Nel pannello di navigazione, scegliere Servizi AWS e scegliere ElastiCache.

Per richiedere un aumento delle quote, consultare [Richiesta di aumento delle quote](#) nella Guida per l'utente di Service Quotas. Se la quota non è ancora disponibile in Service Quotas, utilizza il [modulo di incremento dei limiti](#).

Il tuo account AWS riporta le quote dell'account in relazione a ElastiCache.

Risorsa	Di default
Cache serverless per regione	40
Nodi per regione	300
Nodi per cluster	40
Gruppi di parametri per regione	300
Gruppi di sicurezza per regione	50
Gruppi di sottoreti per regione	300
Sottoreti per gruppo di sottoreti	20

Riferimento

Gli argomenti presenti in questa sezione illustrano l'utilizzo dell'API Amazon ElastiCache e della sezione ElastiCache del AWS CLI. Sono inclusi anche i messaggi di errore e le notifiche di servizio più frequenti.

- [Utilizzo dell'API ElastiCache](#)
- [Informazioni di riferimento ElastiCache](#)
- [Sezione ElastiCache delAWS CLI Riferimento](#)
- [Messaggi di errore Amazon ElastiCache](#)
- [Notifiche](#)

Utilizzo dell'API ElastiCache

Questa sezione fornisce descrizioni delle attività per l'utilizzo e l'implementazione delle operazioni ElastiCache. Per una descrizione completa di queste operazioni, consulta la [Documentazione di riferimento delle API di Amazon ElastiCache](#)

Argomenti

- [Uso dell'API query](#)
- [Librerie disponibili](#)
- [Risoluzione dei problemi delle applicazioni](#)

Uso dell'API query

Parametri di query

Le richieste basate su query HTTP sono richieste HTTP che utilizzano i verbi HTTP GET oppure POST e un parametro di query denominato `Action`.

Ogni richiesta di query deve includere alcuni parametri comuni per gestire l'autenticazione e la selezione di un'azione.

Alcune operazioni accettano elenchi di parametri. Questi elenchi sono specificati usando l'annotazione `param.n`. I valori di `n` sono numero a partire da 1.

Autenticazione delle richieste di query

È possibile inviare richieste di query solo tramite HTTPS ed è necessario includere una firma in ogni richiesta di query. Questa sezione descrive come creare la firma. Il metodo descritto nella seguente procedura è noto come Signature Version 4.

Di seguito sono riportate le fasi di base utilizzate per l'autenticazione delle richieste in AWS. Si presuppone che abbia eseguito la registrazione in AWS e disponga di un ID chiave di accesso e di una chiave di accesso segreta.

Processo di autenticazione delle query

1. Il mittente crea una richiesta in AWS.
2. Il mittente calcola la firma della richiesta, un hash con chiave per il codice di autenticazione di messaggi basati su hash (HMAC) con una funzione hash SHA-1, come definito nella successiva sezione di questo argomento.
3. Il mittente della richiesta invia ad AWS i dati della richiesta, la firma e l'ID chiave di accesso, ovvero l'identificatore per la chiave di accesso segreta in uso.
4. AWS utilizza l'ID chiave di accesso per cercare la chiave di accesso segreta.
5. AWS genera una firma a partire dai dati della richiesta e dalla chiave di accesso segreta utilizzando lo stesso algoritmo con cui viene calcolata la firma nella richiesta.
6. Se le firme corrispondono, la richiesta viene considerata autentica. Se il confronto non va a buon fine, la richiesta viene scartata e AWS invia una risposta di errore.

Note

Se una richiesta contiene un parametro `Timestamp`, la firma calcolata per la richiesta scade 15 minuti dopo il relativo valore.

Se una richiesta contiene un parametro `Expires`, la firma scade in corrispondenza dell'ora specificata dal parametro `Expires`.

Per calcolare la firma della richiesta

1. Creare la stringa di query in forma canonica necessaria successivamente durante questa procedura:

- a. Ordinare i componenti della stringa di query UTF-8 per nome di parametro in base a un ordine naturale dei byte. I parametri possono avere fonte da GET URI o dal corpo POST (quando Content-Type è application/x-www-form-urlencoded).
 - b. Codificare in formato URL il nome e i valori di parametro in base alle seguenti regole:
 - i. Non codificare in formato URL i caratteri non riservati definiti da RFC 3986. I caratteri non riservati sono A-Z, a-z, 0-9, trattino (-), trattino basso (_), punto (.) e tilde (~).
 - ii. Codificare con codifica percentuale tutti gli altri caratteri con %XY, dove X e Y sono caratteri esadecimali (0-9 e A-F maiuscole).
 - iii. Codificare con codifica percentuale i caratteri UTF-8 estesi nel formato %XY%ZA....
 - iv. Codificare con codifica percentuale il carattere di spazio come %20 (non utilizzare +, come negli schemi di codifica comuni).
 - c. Separare i nomi di parametro codificati dai rispettivi valori codificati con il segno di uguale (=), ovvero il carattere ASCII 61, anche se il valore del parametro è vuoto.
 - d. Separare le coppie nome-valore con una E commerciale (&), codice ASCII 38.
2. Creare la stringa di firma in base al seguente esempio di grammatica ("`\n`" rappresenta una nuova riga ASCII).

```
StringToSign = HTTPVerb + "\n" +  
ValueOfHostHeaderInLowercase + "\n" +  
HTTPRequestURI + "\n" +  
CanonicalizedQueryString <from the preceding step>
```

Il componente HTTPRequestURI è il componente del percorso assoluto HTTP dell'URI fino alla stringa di query (senza includerla). Se HTTPRequestURI è vuoto, utilizzare una barra (/).

3. Calcolare una firma HMAC compatibile con RFC 2104 utilizzando la stringa appena creata, la chiave di accesso segreta come chiave e SHA256 o SHA1 come algoritmo hash.

Per ulteriori informazioni, consulta <https://www.ietf.org/rfc/rfc2104.txt>.

4. Convertire il valore risultante in base64.
5. Includere il valore come valore del parametro Signature nella richiesta.

Di seguito è riportata una richiesta di esempio (le interruzioni di riga sono aggiunte per chiarezza).

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DescribeCacheClusters  
&CacheClusterIdentifier=myCacheCluster  
&SignatureMethod=HmacSHA256  
&SignatureVersion=4  
&Version=2014-12-01
```

Per la stringa di query precedente, è necessario calcolare la firma HMAC sulla seguente stringa.

```
GET\n  
elasticache.amazonaws.com\n  
Action=DescribeCacheClusters  
&CacheClusterIdentifier=myCacheCluster  
&SignatureMethod=HmacSHA256  
&SignatureVersion=4  
&Version=2014-12-01  
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
&X-Amz-Credential=AKIADQKE4SARGYLE%2F20140523%2Fus-west-2%2Felasticache  
%2Faws4_request  
&X-Amz-Date=20141201T223649Z  
&X-Amz-SignedHeaders=content-type%3Bhost%3Buser-agent%3Bx-amz-content-sha256%3Bx-  
amz-date  
content-type:  
host:elasticache.us-west-2.amazonaws.com  
user-agent:CacheServicesAPICommand_Client  
x-amz-content-sha256:  
x-amz-date:
```

Il risultato è la richiesta firmata seguente.

```
https://elasticache.us-west-2.amazonaws.com/  
?Action=DescribeCacheClusters  
&CacheClusterIdentifier=myCacheCluster  
&SignatureMethod=HmacSHA256  
&SignatureVersion=4  
&Version=2014-12-01  
&X-Amz-Algorithm=&AWS;4-HMAC-SHA256  
&X-Amz-Credential=AKIADQKE4SARGYLE/20141201/us-west-2/elasticache/aws4_request  
&X-Amz-Date=20141201T223649Z  
&X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-date  
&X-Amz-Signature=2877960fced9040b41b4feaca835fd5cfeb9264f768e6a0236c9143f915ffa56
```

Per informazioni dettagliate sul processo di firma e sul calcolo della firma della richiesta, consulta l'argomento [Processo di firma Signature Version 4](#) e i relativi argomenti secondari.

Librerie disponibili

AWS offre SDK (Software Development Kit) per sviluppatori software che preferiscono creare applicazioni utilizzando API specifiche del linguaggio anziché API di query. Questi SDK forniscono funzioni di base (non incluse nelle API), come autenticazione di richiesta, nuovi tentativi di richiesta e gestione degli errori per iniziare in modo più semplice. Sono disponibili SDK e risorse aggiuntive per i seguenti linguaggi di programmazione:

- [Java](#)
- [Windows e .NET](#)
- [PHP](#)
- [Python](#)
- [Ruby](#)

Per informazioni su altri linguaggi, consulta [Codici di esempio e librerie](#).

Risoluzione dei problemi delle applicazioni

ElastiCache fornisce errori specifici e descrittivi per aiutarti a risolvere i problemi mentre interagisci con l'API ElastiCache.

Errore durante il recupero

In genere, si desidera che l'applicazione verifichi se una richiesta ha generato un errore prima di trascorrere del tempo a elaborare i risultati. Il modo più semplice per determinare se si è verificato un errore consiste nel cercare un nodo `ERROR` nella risposta dall'API ElastiCache.

La sintassi XPath fornisce un modo semplice per cercare la presenza di un `ERROR` nodo, così come un modo semplice per recuperare il codice di errore e il messaggio. Il seguente snippet di codice usa Perl e il modulo `XML::XPath` per determinare se si è verificato un errore durante una richiesta. Se si è verificato un errore, il codice stampa il primo codice di errore e il messaggio nella risposta.

```
use XML::XPath;
```

```
my $xp = XML::XPath->new(xml =>$response);
if ( $xp->find("//Error") )
{print "There was an error processing your request:\n", " Error code: ",
$xml->findvalue("//Error[1]/Code"), "\n", " ",
$xml->findvalue("//Error[1]/Message"), "\n\n"; }
```

Suggerimenti per la risoluzione dei problemi

Si consigliano i seguenti processi per diagnosticare e risolvere i problemi con l'API ElastiCache.

- Verificare che ElastiCache sia eseguito correttamente.

A tale scopo, apri una finestra del browser e invia una richiesta di query al servizio ElastiCache, (ad esempio <https://elasticache.amazonaws.com>). Un'eccezione `MissingAuthenticationTokenException` o un errore server interno 500 conferma che il servizio è disponibile e risponde alle richieste.

- Verificare la struttura della richiesta.

Ogni operazione ElastiCache ha una pagina di riferimento nella documentazione di riferimento dell'API ElastiCache. Controllare nuovamente che si stia usando i parametri correttamente. Per avere delle idee su cosa potrebbe essere sbagliato, guarda le richieste di esempio o gli scenari utente per vedere se quegli esempi stanno eseguendo operazioni simili.

- Controllare il forum.

Per ElastiCache è disponibile un forum di discussione in cui è possibile cercare soluzioni ai problemi che altri utenti hanno riscontrato lungo il percorso. Per visualizzare il forum, accedi all'indirizzo

<https://forums.aws.amazon.com/>

Impostazione dell'interfaccia a riga di comando di ElastiCache

Questa sezione descrive i prerequisiti per ottenere gli strumenti a riga di comando, dove trovarli, come impostare gli strumenti e il loro ambiente; include una serie di esempi comuni di utilizzo degli strumenti.

Le istruzioni indicate in questo argomento vanno seguite solo se si intende utilizzare la AWS CLI di ElastiCache.

Important

L-interfaccia a riga di comando (CLI) di Amazon ElastiCache non supporta alcun miglioramento ElastiCache dopo la versione API 2014-09-30. Per utilizzare la funzionalità di ElastiCache più recente dalla riga di comando, utilizza il comando [Interfaccia a riga di comando di AWS](#).

Argomenti

- [Prerequisiti](#)
- [Ottenere gli strumenti a riga di comando](#)
- [Configurazione degli strumenti](#)
- [Fornire le credenziali per gli strumenti](#)
- [Variabili di ambiente](#)

Prerequisiti

In questo documento si presuppone che si possa lavorare in un ambiente Linux/UNIX o Windows. Gli strumenti a riga di comando di Amazon ElastiCache funzionano anche su Mac OS X, un ambiente basato su UNIX, ma le istruzioni specifiche per Mac OS X non sono incluse in questa guida.

Per convenzione, tutti i testi a riga di comando hanno un prefisso costituito da un prompt generico **PROMPT>** della riga di comando. Il prompt della riga di comando sul tuo computer potrebbe essere diverso. Utilizziamo anche **\$** per indicare un comando specifico Linux/UNIX e **C:\>** per un comando specifico Windows. L'output di esempio derivante dal comando viene visualizzato subito dopo, senza alcun prefisso.

Java Runtime Environment

Per l'esecuzione, gli strumenti a riga di comando utilizzati in questa guida richiedono almeno la versione 5 di Java. È accettabile l'installazione di JRE o di JDK. Per visualizzare e scaricare i JRE per una serie di piattaforme, tra cui Linux/UNIX e Windows, consulta [Java SE Download](#).

Impostazione della variabile Java Home

Gli strumenti a riga di comando dipendono da una variabile di ambiente (JAVA_HOME) per l'individuazione di Java Runtime. Tale variabile deve essere impostata sul percorso completo della

directory contenente una sottodirectory denominata `bin`, che a sua volta contiene il file eseguibile `java` (su Linux e UNIX) o il file eseguibile `java.exe` (su Windows).

Per impostare la variabile Java Home

1. Impostare la variabile Java Home.

- In Linux e UNIX, immettere il seguente comando:

```
$ export JAVA_HOME=<PATH>
```

- In Windows immetti il seguente comando:

```
C:\> set JAVA_HOME=<PATH>
```

2. Confermare l'impostazione del percorso eseguendo il comando `$JAVA_HOME/bin/java -version` e verificando l'output.

- Su Linux/UNIX, l'aspetto dell'output sarà simile al seguente:

```
$ $JAVA_HOME/bin/java -version
java version "1.6.0_23"
Java(TM) SE Runtime Environment (build 1.6.0_23-b05)
Java HotSpot(TM) Client VM (build 19.0-b09, mixed mode, sharing)
```

- Su Windows, l'aspetto dell'output sarà simile al seguente:

```
C:\> %JAVA_HOME%\bin\java -version
java version "1.6.0_23"
Java(TM) SE Runtime Environment (build 1.6.0_23-b05)
Java HotSpot(TM) Client VM (build 19.0-b09, mixed mode, sharing)
```

Ottenere gli strumenti a riga di comando

Gli strumenti a riga di comando sono disponibili come file `.zip` sul [Sito Web degli strumenti per sviluppatori di ElastiCache](#). Tali strumenti sono scritti in Java e includono script di shell per Windows

2000/XP/Vista/Windows 7, Linux/UNIX e Mac OSX. Il file .zip è autonomo e non occorre installarlo; basta scaricarlo e decomprimerlo in una directory sul computer locale.

Configurazione degli strumenti

Gli strumenti a riga di comando dipendono da una variabile di ambiente (`AWS_ELASTICACHE_HOME`) per l'individuazione delle librerie di support. Prima di poter utilizzare gli strumenti, occorre impostare questa variabile. Impostala sul percorso della directory in cui hai decompresso gli strumenti a riga di comando. La directory è denominata `ElastiCacheCli-A.B.nnnn` (A e B sono i numeri della versione, n è il numero della release) e contiene sottodirectory denominate `bin` e `lib`.

Per impostare la variabile di ambiente `AWS_ELASTICACHE_HOME`

- Aprire una finestra a riga di comando e immettere uno dei seguenti comandi per impostare la variabile di ambiente `AWS_ELASTICACHE_HOME`.
- In Linux e UNIX, immettere il seguente comando:

```
$ export &AWS;_ELASTICACHE_HOME=<path-to-tools>
```

- In Windows immetti il seguente comando:

```
C:\> set &AWS;_ELASTICACHE_HOME=<path-to-tools>
```

Per semplificare l'utilizzo degli strumenti, consigliamo di aggiungere la directory `BIN` degli strumenti al `PATH` di sistema. Il resto della guida presuppone che la directory `BIN` si trovi nel percorso di sistema.

Per aggiungere la directory `BIN` degli strumenti al percorso di sistema

- Immettere i seguenti comandi per aggiungere la directory `BIN` degli strumenti al `PATH` di sistema.
- In Linux e UNIX, immettere il seguente comando:

```
$ export PATH=$PATH:$&AWS;_ELASTICACHE_HOME/bin
```

- In Windows immetti il seguente comando:

```
C:\> set PATH=%PATH%;%&AWS;_ELASTICACHE_HOME%\bin
```

Note

Quando si chiude la finestra dei comandi, le variabili di ambiente Windows vengono resettate. Si possono anche impostare in modo definitivo. Per ulteriori informazioni, consulta la documentazione relativa alla versione di Windows in uso.

Note

I percorsi che contengono uno spazio devono essere racchiusi tra virgolette, ad esempio: "C:\Program Files\Java"

Fornire le credenziali per gli strumenti

Per utilizzare gli strumenti a riga di comando occorrono la chiave di accesso AWS e la chiave di accesso segreta fornite con l'account AWS. Puoi ottenerle tramite la riga di comando o da un file di credenziali presente nel sistema locale.

L'implementazione include un file di modello `${AWS_ELASTICACHE_HOME}/credential-file-path.template` che va modificato con le informazioni. Di seguito i contenuti del file di modello:

```
AWSAccessKeyId=<Write your AWS access ID>  
AWSSecretKey=<Write your AWS secret key>
```

Important

In UNIX, limita le autorizzazioni al proprietario del file delle credenziali:

```
$ chmod 600 <the file created above>
```


Con il file di credenziali configurato, è necessario impostare la variabile di ambiente `AWS_CREDENTIAL_FILE` in modo che gli strumenti di ElastiCache possano trovare le informazioni.

Per impostare la variabile di ambiente `AWS_CREDENTIAL_FILE`

1. Imposta la variabile di ambiente :

- Su Linux e UNIX, impostare la variabile di ambiente utilizzando il comando seguente:

```
$ export &AWS;_CREDENTIAL_FILE=<the file created above>
```

- Su Windows, impostare la variabile di ambiente utilizzando il comando seguente:

```
C:\> set &AWS;_CREDENTIAL_FILE=<the file created above>
```

2. Verificare il corretto funzionamento della configurazione tramite il comando seguente:

```
elasticache --help
```

Dovrebbe comparire la pagina di utilizzo di tutti i comandi ElastiCache.

Variabili di ambiente

Le variabili di ambiente possono essere utili per lo script e la configurazione delle impostazioni predefinite o per la loro provvisoria sostituzione.

Oltre alla variabile di ambiente `AWS_CREDENTIAL_FILE`, la maggior parte degli strumenti API inclusi con l'interfaccia a riga di comando di ElastiCache supporta le seguenti variabili:

- `EC2_REGION` la regione AWS da usare.
- `AWS_ELASTICACHE_URL`— L'URL da utilizzare per la chiamata di servizio. Non occorre specificare un endpoint regionale diverso se viene specificata la variabile `EC2_REGION` o se viene trasferito il parametro `--region`.

I seguenti esempi mostrano come impostare la variabile di ambiente `EC2_REGION` per configurare la regione utilizzata dagli strumenti dell'API:

Linux, OS X o Unix

```
$ export EC2_REGION=us-west-1
```

Windows

```
$ set EC2_REGION=us-west-1
```

Messaggi di errore Amazon ElastiCache

Amazon ElastiCache restituisce i messaggi di errore riportati di seguito. Potresti ricevere anche altri messaggi di errore, generati da ElastiCache, diversi servizi AWS o Memcached. Per la descrizione dei messaggi di errore non restituiti da ElastiCache, consulta la documentazione originale dei servizi che li hanno generati.

- [Cluster node quota exceeded](#)
- [Customer's node quota exceeded](#)
- [Insufficient cache cluster capacity](#)

Messaggio di errore: Quota dei nodi di cluster superata. Ogni cluster può annoverare al massimo %n nodi in questa regione."

Causa: hai tentato di creare o modificare un cluster, che, di conseguenza, potrebbe ritrovarsi più di %n nodi.

Soluzione: modifica la richiesta affinché il cluster non annoveri più di %n nodi. Se ti occorrono più di %n nodi, effettua una richiesta apposita tramite il [Modulo di richiesta per nodi di Amazon ElastiCache](#).

Per ulteriori informazioni, consulta [Limiti di Amazon ElastiCache](#) nella Riferimenti generali di Amazon Web Services.

Messaggi di errore: Quota di nodi del cliente superata. Puoi impostare al massimo %n nodi in questa regione . In alternativa, "Hai già raggiunto la quota massima di %s nodi in questa regione".

Causa: in seguito al tentativo di creazione o modifica di un cluster, l'account potrebbe annoverare più di %n nodi tra i cluster della regione.

Soluzione: modifica la richiesta per non superare la quota totale di %n nodi ripartiti tra i cluster della regione dell'account. Se ti occorrono più di %n nodi, effettua una richiesta apposita tramite il [Modulo di richiesta per nodi di Amazon ElastiCache](#).

Per ulteriori informazioni, consulta [Limiti di Amazon ElastiCache](#) nella Riferimenti generali di Amazon Web Services.

Messaggio di errore: InsufficientCacheClusterCapacity

Causa: AWS al momento non dispone di sufficiente capacità on demand per soddisfare la richiesta.

Soluzione::

- Attendi alcuni minuti, quindi invia di nuovo la richiesta; la capacità può variare frequentemente.
- Inviare una nuova richiesta con un numero ridotto di nodi o shard (gruppi di nodi). Ad esempio, se si effettua un'unica richiesta di avvio di 15 nodi, tentare creando tre richieste di cinque nodi oppure 15 richieste di un nodo.
- Se si sta avviando un cluster, inviare una nuova richiesta senza specificare alcuna zona di disponibilità.
- Se si sta avviando un cluster, inviare una nuova richiesta utilizzando un tipo di nodo diverso (che sarà possibile ridimensionare in un secondo momento). Per ulteriori informazioni, consulta [Scalabilità ElastiCache per Memcached](#).

Notifiche

In questo argomento vengono descritte le notifiche ElastiCache che potrebbero essere di tuo interesse. Una notifica è una situazione o un evento nella maggior parte dei casi temporaneo, che dura finché non viene individuata o implementata una soluzione. Le notifiche hanno in genere una data di inizio e una data di risoluzione, dopo la quale la notifica non è più rilevante. Una notifica potrebbe o meno essere rilevante per te. Ti consigliamo una linea guida di implementazione che, se seguita, migliora le prestazioni del cluster.

Le notifiche non annunciano caratteristiche o funzionalità nuove o migliorate di ElastiCache.

Notifiche generali ElastiCache

Attualmente non sono disponibili notifiche ElastiCache in sospenso non specifiche del motore.


Notifiche ElastiCache per Memcached

Di seguito sono elencate le notifiche ElastiCache specifiche per il motore Memcached.

Notifiche ElastiCache specifiche per Memcached

- [Avviso: Crawler Memcached LRU causa errori di segmentazione](#)

Avviso: Crawler Memcached LRU causa errori di segmentazione

 Data di avviso: 28 febbraio 2017

In alcuni casi, il cluster può indicare una situazione di instabilità con un errore di segmentazione nel crawler Memcached LRU. Si tratta di un problema nel motore Memcached presente per un periodo di tempo. Questo errore è diventato evidente in Memcached 1.4.33 quando il Crawler LRU è stato abilitato per impostazione predefinita.

Se si verifica questo errore, è consigliabile disabilitare il Crawler LRU fino alla risoluzione. A tale scopo, utilizza `lru_crawler disable` nella riga di comando o modifica il valore del parametro `lru_crawler` (preferito).

Data di risoluzione:

Risoluzione:

- Versione API: 02-02-2015
- Ultimo aggiornamento della documentazione: 27 novembre 2023

La tabella seguente descrive le modifiche importanti apportate in ogni versione della Guida ElastiCache per l'utente di Memcached ElastiCache dopo marzo 2018. Per ricevere notifiche sugli aggiornamenti della documentazione, puoi sottoscrivere il feed RSS.

Aggiornamenti recenti ElastiCache per Memcached

Modifica	Descrizione	Data
Supporto ElastiCache per Memcached 1.6.22	ElastiCache per Memcached ora supporta Memcached 1.6.22. Include le correzioni di bug cumulative della versione 1.6.18 . Per ulteriori informazioni, consulta Memcached versione 1.6.22 .	10 gennaio 2024
ElastiCache for Memcached ora supporta la creazione di cache serverless	Ora puoi creare le cache serverless che semplificano la gestione delle cache e si dimensionano istantaneamente per supportare le applicazioni più esigenti. Per ulteriori informazioni, consulta Scelta tra le opzioni di implementazione . Come parte di questa funzionalità, sono state aggiunte nuove autorizzazioni a ElastiCacheServiceRolePolicy e AmazonElastiCacheFullAccess per consentire l'associazione di cache serverless con endpoint VPC	27 novembre 2023

gestiti. Inoltre, sono state aggiunte le autorizzazioni per supportare un'esperienza di console modificata utilizzando la policy `AmazonElastiCacheFullAccess`.

[Supporto ElastiCache per Memcached 1.6.17](#)

ElastiCache per Memcached ora supporta Memcached 1.6.17. Include correzioni cumulative dei bug della versione [Memcached 1.6.12](#). Per ulteriori informazioni, consultare [Memcached versione 1.6.17](#).

18 gennaio 2023

[ElastiCache per Memcached ora supporta IPv6](#)

ElastiCache supporta le versioni 4 e 6 del protocollo Internet (IPv4 e IPv6), che consente di configurare il cluster in modo che accetti solo connessioni IPv4, solo connessioni IPv6 o entrambe le connessioni IPv4 e IPv6 (dual-stack). IPv6 è supportato per i carichi di lavoro che utilizzano il motore Memcached dalla versione 1.6.6 in avanti su tutte le istanze sviluppate sul [sistema Nitro](#). Non sono previsti costi aggiuntivi per l'accesso tramite IPv6. ElastiCache Per ulteriori informazioni, consulta [Scelta di un tipo di rete](#).

7 novembre 2022

[ElastiCache per Memcached ora supporta la crittografia in transito](#)

La crittografia in transito è una funzione opzionale che consente di aumentare la sicurezza dei dati nei punti più vulnerabili, ossia quando sono in transito da una posizione all'altra. È supportata dalle versioni di Memcached 1.6.12 e successive. Per ulteriori informazioni, consulta la [crittografia ElastiCache in transito \(TLS\)](#).

26 maggio 2022

[ElastiCache ora supporta PrivateLink](#)

AWS PrivateLink consente di accedere in modo privato alle operazioni ElastiCache API senza un gateway Internet, un dispositivo NAT, una connessione VPN o una connessione AWS Direct Connect. Per ulteriori informazioni, consulta [Amazon ElastiCache API e interfaccia VPC endpoints \(AWS PrivateLink\)](#) per Redis o [ElastiCache Amazon API e interfaccia VPC endpoints \(\)](#) per Memcached.

AWS PrivateLink

24 gennaio 2022

[Support per assegnazione di tag a risorse e chiavi di condizione](#)

ElastiCache ora supporta il tagging per aiutarti a gestire i cluster e altre risorse. ElastiCache Per ulteriori informazioni, consulta [Taggare le risorse](#). ElastiCache ElastiCache introduce anche il supporto per le chiavi di condizione. Puoi specificare le condizioni che determinano il modo in cui una policy IAM viene applicata. Per ulteriori informazioni, consulta [Utilizzo delle chiavi di condizione](#).

7 Aprile 2021

[Support per Memcached 1.6.6.](#)

ElastiCache per Memcached ora supporta Memcached 1.6.6. Include correzioni di bug cumulative dalle versioni Memcached 1.5.16 Per ulteriori informazioni, consulta [Memcached versione 1.6.6](#).

12 novembre 2020

[ElastiCache è ora disponibile su AWS Outposts](#)

[AWS Outposts offre](#) AWS servizi, infrastrutture e modelli operativi nativi praticamente a qualsiasi data center, spazio di co-location o struttura locale. Puoi eseguire la distribuzione ElastiCache su Outposts per configurare, gestire e utilizzare la cache in locale, proprio come faresti nel cloud. Per ulteriori informazioni, consulta [Utilizzo di Outposts](#) per Redis o [Utilizzo di Outposts](#) per Memcached.

8 ottobre 2020

[ElastiCache ora supporta Local Zones](#)

Una zona locale è un'estensione di una AWS regione geograficamente vicina agli utenti. È possibile estendere qualsiasi cloud privato virtuale (VPC) da una AWS regione principale alle Local Zones creando una nuova sottorete e assegnandola a una Local Zone. Per ulteriori informazioni, consulta [Utilizzo di Local Zones](#).

25 settembre 2020

[ElastiCache ora supporta le autorizzazioni a livello di risorsa](#)

Ora puoi limitare l'ambito delle autorizzazioni di un utente specificando le ElastiCache risorse in una policy (IAM). AWS Identity and Access Management Per ulteriori informazioni, consulta [Autorizzazioni a livello di risorsa](#).

12 agosto 2020

[ElastiCache ora supporta l'aggiornamento automatico dei cluster ElastiCache](#)

Amazon ElastiCache ora supporta l'aggiornamento automatico dei ElastiCache cluster dopo la «data di scadenza consigliata» dell'aggiornamento del servizio. ElastiCache utilizza la finestra di manutenzione per pianificare l'aggiornamento automatico dei cluster applicabili. Per ulteriori informazioni, vedere [Aggiornamenti self-service](#).

13 maggio 2020

[Amazon ElastiCache ora supporta i nodi di cache T3 Standard](#)

Ora puoi lanciare i nodi di cache T3 Standard espandibili per uso generico di nuova generazione in Amazon. ElastiCache Le istanze T3-Standard di Amazon EC2 offrono un livello base di prestazioni della CPU con la possibilità di incrementarne l'utilizzo quando necessari o fino all'esaurimento dei crediti accumulati. Per ulteriori informazioni, consulta [Tipi di nodi supportati](#).

12 novembre 2019

[ElastiCache for Memcached ora consente agli utenti di applicare gli aggiornamenti del servizio secondo la propria pianificazione](#)

Con questa funzionalità, puoi scegliere di applicare gli aggiornamenti del servizio disponibili al momento desiderato e non solo durante le finestre di manutenzione. Ciò ridurrà al minimo le interruzioni del servizio, in particolare durante i massimi flussi di lavoro, e ti aiuterà a mantenere la conformità con gli aggiornamenti di sicurezza. Per ulteriori informazioni, consulta la sezione [Aggiornamenti self-service in Amazon ElastiCache](#).

9 ottobre 2019

Supporto ElastiCache per Memcached 1.5.16	ElastiCache per Memcached ora supporta Memcached 1.5.16. Include correzioni di bug cumulative dalle versioni Memcached 1.5.14 e Memcached 1.5.15 . Per ulteriori informazioni, consulta Memcached versione 1.5.16 .	6 settembre 2019
ElastiCache Offerte di istanze riservate standard: pagamento anticipato parziale, pagamento anticipato completo e nessun pagamento anticipato.	Le istanze riservate offrono la flessibilità necessaria per prenotare un' ElastiCache istanza Amazon per un periodo di uno o tre anni in base al tipo di istanza e AWS alla regione. Per ulteriori informazioni, consulta Gestione dei costi con nodi riservati .	18 gennaio 2019
Supporto ElastiCache per Memcached 1.5.10	ElastiCache per Memcached ora supporta Memcached 1.5.10. Include il supporto per i parametri <code>no_modern</code> e <code>inline_ascii_resp</code> . Per ulteriori informazioni, consulta Memcached versione 1.5.10 .	14 novembre 2018

[Guida per l'utente ristrutturata](#)

[La guida ElastiCache utente singola è ora ristrutturata in modo che esistano guide utente separate per Redis \(per Redis User Guide\) e ElastiCache per Memcached \(per Memcached User Guide\).](#) [ElastiCache](#) La struttura della documentazione nella sezione [AWS CLI Command Reference: elasticache](#) e [Amazon ElastiCache API Reference](#) rimane invariata.

20 Aprile 2018

La tabella seguente descrive le modifiche importanti alla Guida ElastiCache per l'utente di Memcached ElastiCache prima di marzo 2018.

Modifica	Descrizione	Data della modifica
Support per Regione Asia Pacifico (Osaka-Locale)	<p>ElastiCache ha aggiunto il supporto per la regione Asia Pacifico (locale di Osaka). La regione Asia Pacifico (Osaka) attualmente supporta un'unica zona di disponibilità e funziona solo tramite invito. Per ulteriori informazioni, consulta gli argomenti seguenti:</p> <ul style="list-style-type: none"> • Regioni supportate • Tipi di nodi di cache supportati 	12 febbraio 2018
Support per l'UE (Parigi).	<p>ElastiCache ha aggiunto il supporto per la regione UE (Parigi). Per ulteriori informazioni, consulta gli argomenti seguenti:</p> <ul style="list-style-type: none"> • Regioni supportate 	18 dicembre 2017

Modifica	Descrizione	Data della modifica
	<ul style="list-style-type: none">• Tipi di nodi di cache supportati	
Support per la regione Cina (Ningxia)	<p>Amazon ElastiCache ha aggiunto il supporto per la regione Cina (Ningxia). Per ulteriori informazioni, consulta gli argomenti seguenti:</p> <ul style="list-style-type: none">• Regioni supportate• Tipi di nodi di cache supportati	11 dicembre 2017
Support per i ruoli collegati ai servizi	<p>Questa versione ElastiCache ha aggiunto il supporto per Service Linked Roles (SLR). Per ulteriori informazioni, consulta gli argomenti seguenti:</p> <ul style="list-style-type: none">• Utilizzo di ruoli collegati ai servizi per Amazon ElastiCache• Configura le tue autorizzazioni (solo nuovi ElastiCache utenti)	7 dicembre 2017
Support per i tipi di nodi R4	<p>Questa versione di supporto ElastiCache aggiuntivo supporta i tipi di nodi R4 in tutte le AWS regioni supportate da ElastiCache. È possibile acquistare i tipi di nodo R4 in modalità on demand o come nodi di cache riservati. Per ulteriori informazioni, consulta gli argomenti seguenti:</p> <ul style="list-style-type: none">• Tipi di nodi di cache supportati• Parametri specifici del tipo di nodo Memcached	20 novembre 2017

Modifica	Descrizione	Data della modifica
Argomento dei modelli di connessione	<p>ElastiCache la documentazione aggiunge un argomento che tratta vari modelli di accesso a un ElastiCache cluster in un Amazon VPC.</p> <p>Per ulteriori informazioni, consulta Modelli di accesso a una cache ElastiCache in un Amazon VPC nella Guida per l'utente di ElastiCache .</p>	24 Aprile 2017
Support per Memcached 1.4.34	<p>ElastiCache aggiunge il supporto alla versione 1.4.34 di Memcached, che incorpora una serie di correzioni alle versioni precedenti di Memcached.</p> <p>Per ulteriori informazioni, vedere le note di rilascio di Memcached 1.4.34 su Memcached su. GitHub</p>	10 aprile 2017
Support per Memcached 1.4.33	<p>ElastiCache aggiunge il supporto per la versione 1.4.33 di Memcached. Per ulteriori informazioni, consulta gli argomenti seguenti:</p> <ul style="list-style-type: none">• Memcached versione 1.4.33• Parametri di Memcached 1.4.33 aggiunti	20 dicembre 2016
Support per la regione Europa occidentale (Londra)	<p>ElastiCache aggiunge il supporto per la regione UE (Londra). Al momento sono supportati solo i tipi di nodi T2 e M4. Per ulteriori informazioni, consulta gli argomenti seguenti:</p> <ul style="list-style-type: none">• Regioni supportate• Tipi di nodi di cache supportati	13 dicembre 2016

Modifica	Descrizione	Data della modifica
Support per la regione Canada (Montreal)	<p>ElastiCache aggiunge il supporto per la regione Canada (Montreal). Attualmente in questa regione sono supportati solo i tipi di nodi M4 e T2. AWS Per ulteriori informazioni, consulta gli argomenti seguenti:</p> <ul style="list-style-type: none">• Regioni supportate• Tipi di nodi di cache supportati	8 dicembre 2016
Support per i tipi di nodi M4 e R3	<p>ElastiCache aggiunge il supporto per i tipi di nodi R3 e M4 nella regione del Sud America (San Paolo) e i tipi di nodi M4 nella regione della Cina (Pechino). Per ulteriori informazioni, consulta gli argomenti seguenti:</p> <ul style="list-style-type: none">• Regioni supportate• Tipi di nodi di cache supportati	1 Novembre 2016
support per la regione Stati Uniti orientali 2 (Ohio)	<p>ElastiCache aggiunge il supporto per la regione Stati Uniti orientali (Ohio) (us-east-2) con i tipi di nodi M4, T2 e R3. Per ulteriori informazioni, consulta gli argomenti seguenti:</p> <ul style="list-style-type: none">• Regioni supportate• Tipi di nodi di cache supportati	17 ottobre 2016
Support per i tipi di nodi M4	<p>ElastiCache aggiunge il supporto per la famiglia di tipi di nodi M4 nella maggior parte delle AWS regioni supportate da ElastiCache. È possibile acquistare i tipi di nodo M4 in modalità on demand o come nodi di cache riservati. Per ulteriori informazioni, consulta gli argomenti seguenti:</p> <ul style="list-style-type: none">• Tipi di nodi di cache supportati• Parametri specifici del tipo di nodo Memcached	3 agosto 2016

Modifica	Descrizione	Data della modifica
Support per la regione di Mumbai	ElastiCache aggiunge il supporto per la regione Asia Pacifico (Mumbai). Per ulteriori informazioni, consulta gli argomenti seguenti: <ul style="list-style-type: none">• Tipi di nodi di cache supportati• Parametri specifici del tipo di nodo Memcached	27 giugno 2016
Support per i tipi di nodi R3	ElastiCache aggiunge il supporto per i tipi di nodi R3 nella regione Cina (Pechino) e nella regione del Sud America (San Paolo). Per ulteriori informazioni, consulta Tipi di nodi di cache supportati .	16 marzo 2016
Accesso ElastiCache tramite una funzione Lambda	È stato aggiunto un tutorial sulla configurazione di una funzione Lambda per ElastiCache l'accesso in un Amazon VPC. Per ulteriori informazioni, consulta Esercitazioni e video relativi a ElastiCache .	12 febbraio 2016
Support per la regione Asia Pacifico (Seoul).	ElastiCache aggiunge il supporto per la regione Asia Pacifico (Seoul) (ap-northeast-2) con i tipi di nodi t2, m3 e r3.	6 gennaio 2016
Support per Memcached 1.4.28.	ElastiCache aggiunge il supporto per la versione 1.4.24 di Memcached e i miglioramenti di Memcached dalla versione 1.4.14. Questa versione supporta la gestione della cache utilizzata meno di recente (LRU) come attività in background, la scelta di jenkins o murmur3 come algoritmo di hashing e nuovi comandi; vanta, inoltre, varie correzioni di bug. Per ulteriori informazioni, consulta le note di rilascio di Memcached .	27 agosto 2015

Modifica	Descrizione	Data della modifica
Support per Auto Discovery di Memcached con PHP 5.6	Questa versione di Amazon ElastiCache aggiunge il supporto per il client Memcached Auto Discovery per PHP versione 5.6. Per ulteriori informazioni, consulta Compilazione del codice fonte per il client del cluster ElastiCache per PHP .	29 luglio 2015
Nuovo argomento: Accesso dall'esterno ElastiCache AWS	Aggiunto un nuovo argomento su come accedere alle ElastiCache risorse dall'esterno AWS. Per ulteriori informazioni, vedere Accesso ElastiCache dall'esterno AWS .	9 luglio 2015
Aggiunta di messaggi relativi alla sostituzione di nodi	<p>ElastiCache aggiunge tre messaggi relativi alla sostituzione pianificata dei nodi, ElastiCache: NodeReplacementScheduled, ElastiCache: NodeReplacementRescheduled, ElastiCache: NodeReplacementCanceled.</p> <p>Per ulteriori informazioni e azioni che puoi intraprendere quando è pianificata la sostituzione di un nodo, ElastiCache consulta Notifiche di eventi Amazon SNS.</p>	11 giugno 2015
Support per i tag di allocazione dei costi	ElastiCache aggiunge il supporto per i tag di allocazione dei costi. Per ulteriori informazioni, consulta Monitoraggio dei costi con i tag di allocazione dei costi .	9 febbraio 2015
Supporto per la AWS GovCloud regione (Stati Uniti occidentali)	ElastiCache aggiunge il supporto per la regione AWS GovCloud (Stati Uniti occidentali) (us-gov-west-1).	29 gennaio 2015
Support per la regione Europa (Francoforte)	ElastiCache aggiunge il supporto per la regione Europa (Francoforte) (eu-central-1).	19 gennaio 2015

Modifica	Descrizione	Data della modifica
AWS CloudTrail è supportata la registrazione delle chiamate API	ElastiCache aggiunge il supporto per l'utilizzo AWS CloudTrail per registrare tutte le chiamate ElastiCache API. Per ulteriori informazioni, consulta Registrazione delle chiamate all'API Amazon ElastiCache con AWS CloudTrail .	15 settembre 2014
Nuove dimensioni d'istanza supportate	ElastiCache aggiunge il supporto per ulteriori istanze General Purpose (T2). Per ulteriori informazioni, consulta Configurazione dei parametri di motore con i gruppi di parametri .	11 settembre 2014
Supportato il posizionamento di nodi flessibile per Memcached	ElastiCache aggiunge il supporto per la creazione di nodi Memcached su più zone di disponibilità.	23 luglio 2014
Nuove dimensioni d'istanza supportate	ElastiCache aggiunge il supporto per ulteriori istanze General Purpose (M3) e istanze Memory Optimized (R3). Per ulteriori informazioni, consulta Configurazione dei parametri di motore con i gruppi di parametri .	1 luglio 2014
Individuazione automatica di PHP	Aggiunta del supporto per l'individuazione automatica di PHP versione 5.5.	13 maggio 2014
Support per Amazon Virtual Private Cloud (VPC) di default.	In questa versione, ElastiCache è completamente integrato con Amazon Virtual Private Cloud (VPC). I cluster di cache dei nuovi clienti vengono creati, per impostazione predefinita, in un Amazon VPC . Per ulteriori informazioni, consulta Amazon VPC e sicurezza di ElastiCache .	8 gennaio 2013

Modifica	Descrizione	Data della modifica
Support di PHP per l'individuazione automatica di nodi di cache	L'individuazione automatica dei nodi di cache, nella versione iniziale, offre support per i programmi Java. In questa versione, ElastiCache porta il supporto per l'individuazione automatica dei nodi di cache in PHP.	2 gennaio 2013
Support per Amazon Virtual Private Cloud (VPC)	In questa versione, ElastiCache i cluster possono essere lanciati in Amazon Virtual Private Cloud (VPC). Per impostazione predefinita, i cluster di cache dei nuovi clienti vengono creati automaticamente in un Amazon VPC; i clienti preesistenti possono passare all'Amazon VPC in qualsiasi momento, in base alle proprie esigenze. Per ulteriori informazioni, consulta Amazon VPC e sicurezza di ElastiCache .	20 dicembre 2012
Individuazione automatica dei nodi di cache e nuova versione del motore della cache	<p>ElastiCache fornisce l'individuazione automatica dei nodi di cache, ovvero la possibilità per i programmi client di determinare automaticamente tutti i nodi di cache in un cluster e di avviare e mantenere le connessioni a tutti questi nodi.</p> <p>Questa release offre, inoltre, una nuova versione del motore della cache: Memcached 1.4.14. Questo motore della cache inedito vanta, rispetto al passato, caratteristiche ottimizzate di ribilanciamento dell'allocazione, nonché notevoli miglioramenti in termini di prestazioni e scalabilità e varie correzioni di bug. Mette a disposizione, inoltre, molti nuovi parametri della cache configurabili. Per ulteriori informazioni, consulta Configurazione dei parametri di motore con i gruppi di parametri.</p>	28 Novembre 2012
Nuovi tipi di nodo di cache	Questa versione offre quattro nuovi tipi di nodi di cache.	13 Novembre 2012

Modifica	Descrizione	Data della modifica
Nodi di cache riservati	In questa versione è stato aggiunto il support per i nodi di cache riservati.	5 Aprile 2012
Nuova guida	Questa è la prima versione di Amazon ElastiCache User Guide.	22 agosto 2011

AWS Glossario

Per la AWS terminologia più recente, consultate il [AWS glossario](#) nella sezione Reference. Glossario AWS

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.