



Guida per gli sviluppatori

Flusso di video Amazon Kinesis



Flusso di video Amazon Kinesis: Guida per gli sviluppatori

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 Kinesis Video Streams?	1
Disponibilità nelle Regioni	2
Sei il primo utente di Kinesis Video Streams?	3
Requisiti di sistema	5
Requisiti della fotocamera	5
Sistemi operativi testati	6
Requisiti di archiviazione SDK	6
Come funziona	7
Producer Library e API	8
API Kinesis Video Streams	8
Modello di rilevamento degli endpoint	11
Librerie Producer	11
Video Playback (Riproduzione video)	11
Requisiti di riproduzione	13
Riproduzione video con HLS	17
Riproduzione video con MPEG-DASH	27
Utilizzo dei metadati di streaming	31
Aggiungere metadati a un flusso video Kinesis	32
Utilizzo dei metadati incorporati in un flusso video Kinesis	34
Limiti dei metadati di streaming	35
Modello di dati	36
Elementi di intestazione dello stream	36
Trasmetti i dati della traccia	42
Elementi dell'intestazione del frame	43
Dati del frame MKV	44
Nozioni di base	45
Configurazione di un account	45
Registrati per un Account AWS	46
Crea un utente con accesso amministrativo	46
Crea una Account AWS chiave	47
Creare un flusso video Kinesis	48
Crea un flusso video utilizzando la console	48
Crea un flusso video utilizzando il AWS CLI	48
Inviare dati a un flusso video di Amazon Kinesis	49

Crea l'SDK e gli esempi	49
Esegui gli esempi per caricare contenuti multimediali su Kinesis Video Streams	53
Rivedi gli oggetti di riconoscimento	55
Consuma dati multimediali	55
Visualizza i contenuti multimediali nella console	55
Consuma i dati multimediali utilizzando HLS	56
Agente Edge	57
Operazioni dell'API Amazon Kinesis Video Streams Edge Agent	58
Monitoraggio dell'agente Amazon Kinesis Video Streams Edge	58
Implementa in modalità diversa AWS IoT Greengrass	58
1. Installare le dipendenze	59
2. Crea risorse per gli URL RTSP della tua telecamera IP	60
3. Crea una politica di autorizzazioni IAM	62
4. Creazione di un ruolo IAM	64
5. Creare l'alias del AWS IoT ruolo	65
6. Crea la AWS IoT politica	66
7. Crea AWS IoT qualcosa e ottieni AWS IoT Core le credenziali	68
8. Crea ed esegui Amazon Kinesis Video Streams Edge Agent	71
9. (Facoltativo) Installa l' CloudWatch agente	82
10. (Facoltativo) Esegui Amazon Kinesis Video Streams Edge Agent	85
Esegui la distribuzione su AWS IoT Greengrass	87
1. Crea un'istanza di Ubuntu	88
2. Configura il dispositivo AWS IoT Greengrass principale	89
3. Crea risorse per gli URL RTSP della tua telecamera IP	91
4. Aggiungere le autorizzazioni al ruolo TES	93
5. Installa il componente Secret Manager	95
6. Implementa Amazon Kinesis Video Streams Edge Agent sul dispositivo	98
7. (Facoltativo) Installa il componente di gestione dei AWS IoT Greengrass registri	107
Domande frequenti	111
Quali sistemi operativi supporta Amazon Kinesis Video Streams Edge Agent?	111
L'agente Amazon Kinesis Video Streams Edge supporta i file multimediali H.265?	111
L'agente Amazon Kinesis Video Streams Edge funziona in AL2?	111
Come posso eseguire più stream all'interno dell'oggetto o del AWS IoT dispositivo?	112
Come posso modificare un messaggio StartEdgeConfigurationUpdate dopo che è stato inviato?	112
Hai qualche esempio in comuneScheduleConfigs?	112

Esiste un limite massimo di streaming?	113
Come posso riavviare un processo che ha avuto un errore?	113
Come posso monitorare lo stato del mio agente Amazon Kinesis Video Streams Edge?	114
Streaming di video tramite un VPC	115
Informazioni aggiuntive	115
Procedure per gli endpoint VPC	115
Immagini	118
Nozioni di base su GetImages	118
Guida introduttiva alla consegna di Amazon S3	118
UpdateImageGenerationConfiguration	119
DescribeImageGenerationConfiguration	121
Etichette MKV del produttore	122
Aggiungere tag di metadati in Producer SDK utilizzandoPutEventMetaData	123
Limiti	123
Metadati degli oggetti S3	123
Percorso dell'oggetto S3 (immagine)	124
Raccomandazioni sugli URI di Amazon S3 per proteggersi dalle limitazioni	124
Notifiche	126
UpdateNotificationConfiguration	126
DescribeNotificationConfiguration	126
Tag MKV del produttore	122
Sintassi per i tag MKV di Producer	122
Limiti dei tag MKV	127
.....	127
.....	127
Payload tematico Amazon SNS	127
Visualizzazione dei messaggi Amazon SNS	128
Sicurezza	130
Protezione dei dati	131
Cos'è la crittografia lato server per Kinesis Video Streams?	131
Considerazioni su costi, regioni e prestazioni	131
Come posso iniziare a usare la crittografia lato server?	132
Creazione e utilizzo di una chiave gestita dal cliente	133
Autorizzazioni per l'utilizzo di una chiave gestita dal cliente	133
Controllo dell'accesso alle risorse Kinesis Video Streams tramite IAM	135
Sintassi delle policy	136

Azioni per Kinesis Video Streams	137
Amazon Resource Names (ARN) per Kinesis Video Streams	137
Concedere ad altri account IAM l'accesso a un flusso video Kinesis	138
Policy di esempio	141
Controllo dell'accesso alle risorse Kinesis Video Streams tramite AWS IoT	143
AWS IoT ThingName come nome dello stream	144
AWS IoT CertificateId come nome dello stream	150
Usa AWS IoT le credenziali per eseguire lo streaming su un nome di stream codificato	152
Monitoraggio	153
Convalida della conformità	153
Resilienza	154
Sicurezza dell'infrastruttura	154
Best practice di sicurezza	155
Implementazione dell'accesso con privilegi minimi	155
Uso di ruoli IAM	155
Utilizzalo CloudTrail per monitorare le chiamate API	156
Producer Library	157
Cliente di Kinesis Video Streams Producer	157
Libreria dei produttori di Kinesis Video Streams	158
Argomenti correlati	159
Java Producer Library	159
Procedura: utilizzo dell'SDK Java Producer	159
Fase 1: download e configurazione del codice	161
Fase 2: Scrivere ed esaminare il codice	162
Passaggio 3: Esegui e verifica il codice	163
Android Producer Library	164
Procedura: utilizzo dell'SDK del producer Android	164
Prerequisiti	165
Fase 1: download e configurazione del codice	168
Fase 2: Esamina il codice	170
Passaggio 3: Esegui e verifica il codice	172
C++ Producer Library	173
Modello a oggetti	173
Inserimento di contenuti multimediali nello stream	174
Interfacce di callback	174
Procedura: utilizzo dell'SDK del producer C++	175

Fase 1: download e configurazione del codice	176
Fase 2: Scrivere ed esaminare il codice	176
Passaggio 3: Esegui e verifica il codice	183
Usare C++ Producer SDK come plugin GStreamer	184
Utilizzo del C++ Producer SDK come plugin GStreamer in un contenitore Docker	184
Uso del logging	184
C Producer Library	185
Modello a oggetti	185
Inserimento di contenuti multimediali nello stream	186
Procedura: utilizzo dell'SDK del producer C	187
Fase 1: Download del codice	188
Fase 2: Scrivere ed esaminare il codice	189
Passaggio 3: Esegui e verifica il codice	192
SDK C++ Producer su Raspberry Pi	193
Prerequisiti	194
Crea un utente IAM con il permesso di scrivere su Kinesis Video Streams	195
Collega il tuo Raspberry Pi alla tua rete Wi-Fi	196
Connect da remoto al tuo Raspberry Pi	197
Configura la fotocamera Raspberry Pi	197
Installa i prerequisiti del software	198
Scarica e crea l'SDK Kinesis Video Streams C++ Producer	199
Trasmetti video al tuo stream video Kinesis e guarda il live streaming	200
Documentazione di riferimento	201
Limiti di Producer SDK	201
Riferimento al codice di errore	205
Flag di adattamento del NAL	262
Strutture del producer	263
Strutture di flussi	265
Callback	285
Libreria parser del flusso	293
Procedura: utilizzo della libreria Kinesis Video Stream Parser	293
Prerequisiti	293
Passaggio 1: scarica e configura il codice	294
Approfondimenti	294
Passaggio 2: scrivere ed esaminare il codice	294
StreamingMkvReader	295

FragmentMetadataVisitor	295
OutputSegmentMerger	297
KinesisVideoExample	298
Approfondimenti	301
Fase 3: Esegui e verifica il codice	301
Esempi	303
Esempi: invio di dati a Kinesis Video Streams	303
Esempi: recupero di dati da Kinesis Video Streams	303
Esempi: riproduzione di dati video	303
Prerequisiti	303
Plugin GStreamer - kvssink	304
Scarica, crea e configura l'elemento GStreamer	305
Esegui l'elemento GStreamer	305
Comandi di avvio	306
Esegui l'elemento GStreamer in un contenitore Docker	308
Parametri del server	311
PutMedia API	325
Passaggio 1: scarica e configura il codice	326
Passaggio 2: scrivere ed esaminare il codice	327
Passaggio 3: Esegui e verifica il codice	329
RTSP e Docker	330
Tutorial video	330
Prerequisiti	331
Crea l'immagine Docker	331
Esegui l'applicazione di esempio RTSP	332
Renderer	333
Prerequisiti	334
Esecuzione dell'esempio del renderer	334
Come funziona	335
Monitoraggio	337
Monitoraggio dei parametri con CloudWatch	337
CloudWatch indicazioni sulle metriche	353
Monitoraggio dell'agente Amazon Kinesis Video Streams Edge con CloudWatch	357
CloudWatch indicazioni sui parametri per Amazon Kinesis Video Streams Edge Agent	360
Registrazione delle chiamate API di CloudTrail con	362
Amazon Kinesis Video Streams e CloudTrail	363

Esempio: voci dei file di registro di Amazon Kinesis Video Streams	364
Quote	368
Quote di servizio API del piano di controllo	368
Quote dei servizi API multimediali e archiviati	373
Framment-metadati e framment-media quote	378
Quote di metadati per frammenti	381
Tag di streaming	382
Risoluzione dei problemi	383
Problemi generali	383
Latenza troppo elevata	383
Problemi relativi alle API	384
Errore: "Opzioni sconosciute"	384
Errore: "Unable to determine service/operation name to be authorized" (Impossibile determinare il nome del servizio/operazione da autorizzare)	384
Errore: "Failed to put a frame in the stream" (Impossibile inserire un fotogramma nel flusso)	385
Errore: «Il servizio ha chiuso la connessione prima della ricezione del messaggio finale AckEvent »	385
Errore "STATUS_STORE_OUT_OF_MEMORY"	386
Problemi HLS	386
Problemi con Java	386
Abilitazione dei log di Java	386
Problemi con Producer Library	387
Impossibile compilare l'SDK producer	388
Il flusso video non viene visualizzato nella console	388
Errore "Security token included in the request is invalid" durante lo streaming dei dati con l'applicazione demo GStreamer	389
Errore "Failed to submit frame to Kinesis Video client"	389
L'applicazione GStreamer viene arrestata con il messaggio "streaming stopped, reason not-negotiated" su OS X	389
Errore "Failed to allocate heap" durante la creazione del client Kinesis Video nella demo GStreamer su Raspberry Pi	390
Errore "Illegal Instruction" durante l'esecuzione della demo GStreamer su Raspberry Pi	390
La videocamera non viene caricata su Raspberry Pi	390
Impossibile trovare la videocamera su macOS High Sierra	391
File jni.h non trovato durante la compilazione su macOS High Sierra	391

Errori di curl durante l'esecuzione dell'applicazione demo GStreamer	391
Assertione timestamp/intervallo in fase di runtime su Raspberry Pi	391
Assertione su <code>gst_value_set_fraction_range_full</code> su Raspberry Pi	392
STATUS_MKV_INVALID_ANNEXB_NALU_IN_FRAME_DATA (0x3200000d) error on Android	392
È stata raggiunta la durata massima del frammento (errore)	392
Errore "Invalid thing name passed (Passato nome di cosa non valido)" quando si utilizza l'autorizzazione IoT	393
Problemi relativi alla libreria Stream Parser	393
Impossibile accedere a un singolo fotogramma del flusso	393
Errore di decodifica frammento	394
Problemi di rete	394
Cronologia dei documenti	395
Documentazione di riferimento delle API	400
Operazioni	400
Flusso di video Amazon Kinesis	401
Amazon Kinesis Video Streams	524
Amazon Kinesis Video Streams	541
Flusso di video Amazon Kinesis	589
Amazon Kinesis Video Video Video Video Video Video Video Video WebRTC Video Video Video Video Video Video Video Video	598
Tipi di dati	602
Flusso di video Amazon Kinesis	604
i supporti per Amazon Kinesis Video Streams	644
Amazon Kinesis Video Streams	647
Flusso di video Amazon Kinesis	665
Amazon Kinesis Video Video Video Video Video Video Video Video WebRTC Video Video Video Video Video Video Video Video	667
Errori comuni	667
Parametri comuni	669
.....	dclxxii

Che cos'è Kinesis Video Streams?

Puoi utilizzare Amazon Kinesis Video Streams, un servizio Servizio AWS completamente gestito, per trasmettere video in diretta dai dispositivi a o creare applicazioni Cloud AWS per l'elaborazione video in tempo reale o l'analisi video orientata ai batch.

Kinesis Video Streams non è solo spazio di archiviazione per dati video. ma anche a visualizzare i flussi video in tempo reale nel momento stesso in cui vengono ricevuti nel cloud. Puoi monitorare i tuoi live streaming in o sviluppare la AWS Management Console tua applicazione di monitoraggio che utilizzi la libreria API Kinesis Video Streams per visualizzare video in diretta.

Puoi utilizzare Kinesis Video Streams per acquisire enormi quantità di dati video in diretta da milioni di fonti, tra cui smartphone, telecamere di sicurezza, webcam, telecamere integrate in auto, droni e altre fonti. Puoi anche inviare dati non video con serializzazione temporale come dati audio, immagini termiche, dati di profondità e dati RADAR. Trasmettendo video in diretta da queste fonti a un flusso video Kinesis, puoi creare applicazioni per accedere ai dati in tempo reale per l' frame-by-frameelaborazione a bassa latenza. Kinesis Video Streams è indipendente dalla fonte. Puoi trasmettere video in streaming dalla webcam di un computer utilizzando la [Plugin GStreamer - kvssink](#) libreria o da una videocamera in rete utilizzando il protocollo di streaming in tempo reale (RTSP).

Puoi anche configurare il tuo flusso video Kinesis per archiviare in modo duraturo i dati multimediali per il periodo di conservazione specificato. Kinesis Video Streams archivia automaticamente questi dati e li crittografa quando sono inattivi. Inoltre, Kinesis Video Streams indicizza i dati archiviati in base sia ai timestamp del produttore che ai timestamp di ingestione. È possibile creare applicazioni che elaborano periodicamente in batch i dati video oppure creare applicazioni che richiedono l'accesso una tantum ai dati storici per diversi casi d'uso.

Le tue applicazioni personalizzate, in tempo reale o orientate ai batch, possono essere eseguite su istanze Amazon EC2. Queste applicazioni possono elaborare i dati utilizzando algoritmi open source di deep learning o utilizzare applicazioni di terze parti che si integrano con Kinesis Video Streams.

I vantaggi dell'utilizzo di Kinesis Video Streams includono quanto segue:

- Connettiti e trasmetti in streaming da milioni di dispositivi: puoi utilizzare Kinesis Video Streams per connettere e trasmettere video, audio e altri dati da milioni di dispositivi, tra cui smartphone di consumo, droni e dash cam. Puoi utilizzare le librerie per produttori di Kinesis Video Streams per

configurare i tuoi dispositivi e trasmettere in modo affidabile in tempo reale o come caricamenti multimediali. after-the-fact

- Archivia, crittografa e indicizza i dati in modo duraturo: puoi configurare il tuo flusso video Kinesis per archiviare in modo duraturo i dati multimediali per periodi di conservazione personalizzati. Kinesis Video Streams genera anche un indice sui dati archiviati basato su timestamp generati dal produttore o dal lato del servizio. Le tue applicazioni possono recuperare dati specifici in un flusso utilizzando l'indice temporale.
- Concentrati sulla gestione delle applicazioni anziché dell'infrastruttura: Kinesis Video Streams è serverless, quindi non c'è alcuna infrastruttura da configurare o gestire. Non devi preoccuparti dell'implementazione, della configurazione o della scalabilità elastica dell'infrastruttura sottostante perché i flussi di dati e il numero di applicazioni che consumano crescono e si riducono. Kinesis Video Streams esegue automaticamente tutta l'amministrazione e la manutenzione necessarie per gestire gli stream, in modo che tu possa concentrarti sulle applicazioni, non sull'infrastruttura.
- Crea applicazioni in tempo reale e in batch su flussi di dati: puoi utilizzare Kinesis Video Streams per creare applicazioni personalizzate in tempo reale che operano su flussi di dati live e creare applicazioni batch o una tantum che funzionano su dati persistenti in modo duraturo senza requisiti di latenza rigorosi. Puoi creare, distribuire e gestire applicazioni personalizzate: soluzioni open source (Apache MXNet, OpenCV), sviluppate internamente o di terze parti che utilizzano per elaborare e analizzare i tuoi stream. Marketplace AWS Puoi utilizzare le API Kinesis Get Video Streams per creare più applicazioni simultanee che elaborano i dati in tempo reale o in batch.
- Trasmetti i dati in modo più sicuro: Kinesis Video Streams crittografa tutti i dati mentre fluiscono attraverso il servizio e quando i dati persistono. Kinesis Video Streams applica la crittografia basata su Transport Layer Security (TLS) sullo streaming di dati dai dispositivi e crittografa tutti i dati inattivi utilizzando (). AWS Key Management Service AWS KMS Inoltre, puoi gestire l'accesso ai tuoi dati utilizzando (IAM). AWS Identity and Access Management
- Pagamento in base al consumo: per ulteriori informazioni, consulta [AWS Pricing Calculator](#).

Disponibilità nelle Regioni

Amazon Kinesis Video Streams è disponibile nelle seguenti regioni:

Nome della regione	AWSCodice regionale
Stati Uniti orientali (Ohio)	us-east-2

Nome della regione	AWSCodice regionale
US East (N. Virginia)	us-east-1
US West (Oregon)	us-west-2
Africa (Cape Town)	af-south-1
Asia Pacifico (Hong Kong)	ap-east-1
Asia Pacific (Mumbai)	ap-south-1
Asia Pacific (Seoul)	ap-northeast-2
Asia Pacific (Singapore)	ap-southeast-1
Asia Pacific (Sydney)	ap-southeast-2
Asia Pacific (Tokyo)	ap-northeast-1
Canada (Central)	ca-central-1
China (Beijing)	cn-north-1
Europe (Frankfurt)	eu-central-1
Europe (Ireland)	eu-west-1
Europe (London)	eu-west-2
Europe (Paris)	eu-west-3
Sud America (San Paolo)	sa-east-1

Sei il primo utente di Kinesis Video Streams?

Se utilizzi Kinesis Video Streams per la prima volta, ti consigliamo di leggere le seguenti sezioni nell'ordine:

1. [Kinesis Video Streams: come funziona](#)— Per conoscere i concetti di Kinesis Video Streams.

2. [Guida introduttiva ad Amazon Kinesis Video Streams](#)— Per configurare il tuo account e testare Kinesis Video Streams.
3. [Librerie per produttori di Kinesis Video Streams](#)— Per saperne di più sulla creazione di un'applicazione per produttori Kinesis Video Streams.
4. [Libreria di analisi dei flussi video Kinesis](#)— Per saperne di più sull'elaborazione dei frame di dati in entrata in un'applicazione consumer Kinesis Video Streams.
5. [Esempi di Amazon Kinesis Video Streams](#)— Per vedere altri esempi di cosa puoi fare con Kinesis Video Streams.

Requisiti di sistema di Kinesis Video Streams

Le seguenti sezioni contengono i requisiti hardware, software e di storage per Amazon Kinesis Video Streams.

Argomenti

- [Requisiti della fotocamera](#)
- [Sistemi operativi testati](#)
- [Requisiti di archiviazione SDK](#)

Requisiti della fotocamera

Le videocamere utilizzate per eseguire l'SDK e gli esempi di Kinesis Video Streams Producer hanno i seguenti requisiti di memoria:

- La visualizzazione dei contenuti SDK richiede 16 MB di memoria.
- La configurazione predefinita dell'applicazione di esempio è di 128 MiB di memoria. Si tratta di un valore appropriato per i producer con una buona connettività di rete e senza requisiti di buffering aggiuntivo. Se la connessione di rete è scadente ed è necessario ulteriore buffering, è possibile calcolare la memoria richiesta per ogni secondo di buffering moltiplicando il frame rate al secondo per le dimensioni di memoria del fotogramma. Per ulteriori informazioni sull'allocazione della memoria, consultare [StorageInfo](#).

È consigliabile usare videocamere USB o RTSP (Real Time Streaming Protocol) in grado di codificare i dati tramite H.264: così si evita che sia la CPU a dover eseguire la crittografia.

Attualmente, l'applicazione demo non supporta lo User Datagram Protocol (UDP) per lo streaming RTSP. Questa funzionalità verrà aggiunta in futuro.

L'SDK producer supporta i seguenti tipi di videocamere:

- Videocamere Web
- Videocamere USB
- Videocamere con codifica H.264 (preferite)
- Videocamere senza codifica H.264

- Modulo telecamera Raspberry Pi. Si tratta dell'opzione preferita per i dispositivi Raspberry Pi perché si connette alla GPU per il trasferimento di dati video senza includere l'elaborazione da parte della CPU.
- Videocamere RTSP (di rete). Queste videocamere sono preferibili perché i flussi video sono già codificati con H.264.

Sistemi operativi testati

Le videocamere Web e RTSP sono state testate con i seguenti dispositivi e sistemi operativi:

- Mac mini
 - High Sierra
- MacBook Computer portatili Pro
 - Sierra (10.12)
 - El Capitan (10.11)
- Laptop HP con Ubuntu 16.04
- Ubuntu 17.10 (container Docker)
- Raspberry Pi 3

Requisiti di archiviazione SDK

Per l'installazione delle [Librerie per produttori di Kinesis Video Streams](#) è richiesta una capacità di storage minima di 170 MB e sono raccomandati 512 MB.

Kinesis Video Streams: come funziona

Argomenti

- [Supporto per l'API Kinesis Video Streams e le librerie dei produttori](#)
- [Riproduzione di Kinesis Video Streams](#)
- [Utilizzo dei metadati di streaming con Kinesis Video Streams](#)
- [Modello di dati Kinesis Video Streams](#)

Puoi utilizzare Amazon Kinesis Video Streams, un programma Servizio AWS completamente gestito, per trasmettere video in diretta dai dispositivi e archivarli in Cloud AWS modo duraturo. Puoi quindi creare le tue applicazioni per l'elaborazione di video in tempo reale o eseguire analisi dei video in batch.

Il diagramma seguente fornisce una panoramica del funzionamento di Kinesis Video Streams.

Il diagramma mostra l'interazione tra i componenti seguenti:

- **Produttore:** qualsiasi fonte che inserisce dati in un flusso video Kinesis. Un produttore può essere qualsiasi dispositivo di generazione video, come una telecamera di sicurezza, una videocamera indossabile, una fotocamera per smartphone o una telecamera da cruscotto. Un producer può anche inviare dati non video, come feed audio, immagini o dati RADAR.

Un producer singolo può generare uno o più flussi video. Ad esempio, una videocamera può inviare dati video a un flusso video Kinesis e dati audio a un altro.

- **Librerie Kinesis Video Streams Producer:** un set di software e librerie che puoi installare e configurare sui tuoi dispositivi. Puoi utilizzare queste librerie per connetterti in modo sicuro e trasmettere video in streaming in modo affidabile in diversi modi, anche in tempo reale, dopo averli memorizzati nel buffer per alcuni secondi o come caricamenti multimediali. after-the-fact
- **Kinesis video stream:** una risorsa che puoi utilizzare per trasportare dati video in diretta, archivarli facoltativamente e renderli disponibili per il consumo sia in tempo reale che in batch o una tantum. In una configurazione tipica, uno stream video Kinesis include un solo produttore che pubblica dati al suo interno.

Il flusso è in grado di trasportare audio e flussi di dati simili codificati temporalmente, ad esempio feed di rilevamento profondità, feed RADAR e molto altro. Puoi creare un flusso video Kinesis utilizzando AWS Management Console o utilizzando gli SDK a livello di codice. AWS

Più applicazioni indipendenti possono utilizzare un flusso video Kinesis in parallelo.

- **Consumatore:** recupera dati, come frammenti e frame, da uno stream video Kinesis per visualizzarli, elaborarli o analizzarli. Generalmente questi utenti sono denominati applicazioni Kinesis Video Streams. È possibile scrivere applicazioni che consumano ed elaborano dati in Kinesis Video Streams in tempo reale o dopo l'archiviazione e l'indicizzazione temporale dei dati quando non è richiesta un'elaborazione a bassa latenza. Puoi creare queste applicazioni consumer da eseguire su istanze Amazon EC2.
- [Libreria di analisi dei flussi video Kinesis](#)— Consente alle applicazioni Kinesis Video Streams di ottenere contenuti multimediali in modo affidabile da Kinesis Video Stream a bassa latenza. Inoltre, analizza i limiti del frame nel contenuto multimediale in modo che le applicazioni possano concentrarsi sull'elaborazione e l'analisi dei frame stesso.

Supporto per l'API Kinesis Video Streams e le librerie dei produttori

Kinesis Video Streams fornisce API per creare e gestire stream e leggere o scrivere dati multimediali da e verso uno stream. La console Kinesis Video Streams, oltre alle funzionalità di amministrazione, supporta video-on-demand anche il live e la riproduzione. Kinesis Video Streams fornisce anche una serie di librerie di produttori che è possibile utilizzare nel codice dell'applicazione per estrarre dati dalle fonti multimediali e caricarli nel flusso video Kinesis.

Argomenti

- [API Kinesis Video Streams](#)
- [Modello di rilevamento degli endpoint](#)
- [Librerie Producer](#)

API Kinesis Video Streams

Kinesis Video Streams fornisce API per la creazione e la gestione di Kinesis Video Streams. Fornisce inoltre API per la lettura e la scrittura di dati multimediali su un flusso, nel modo seguente:

- **Producer API** — Kinesis Video Streams `PutMedia` fornisce un'API per scrivere dati multimediali su un flusso video Kinesis. In una richiesta `PutMedia`, il producer invia un flusso di frammenti multimediali. Per frammento si intende una sequenza autonoma di frame. I frame appartenenti a un frammento non devono avere alcun tipo di dipendenza dai frame di altri frammenti. Per ulteriori informazioni, consulta [PutMedia](#).

All'arrivo dei frammenti, Kinesis Video Streams assegna un numero di frammento univoco, in ordine crescente. Memorizza anche i timestamp lato produttore e lato server per ogni frammento, come metadati specifici di Kinesis Video Streams.

- **API per i consumatori:** i consumatori possono utilizzare le seguenti API per ottenere dati da uno stream:
 - **GetMedia:** quando si utilizza questa API, i consumer devono identificare il frammento di partenza. L'API restituisce i frammenti nell'ordine in cui sono stati aggiunti nel flusso (in ordine crescente per numero di frammento). I dati multimediali nel frammento vengono compressi in un formato strutturato, ad esempio [Matroska \(MKV\)](#). Per ulteriori informazioni, consulta [GetMedia](#).

Note

`GetMedia` sa dove sono i frammenti (archiviati nello datastore o disponibili in tempo reale). Ad esempio, se `GetMedia` determina che i frammenti di partenza sono archiviati, ne avvia la restituzione dal datastore. Quando deve restituire frammenti più recenti che non sono ancora stati archiviati, `GetMedia` passa alla lettura dei frammenti da un buffer di flusso in memoria.

Questo è un esempio di un consumer continuo, che elabora i frammenti nell'ordine in cui vengono acquisiti dal flusso.

`GetMedia` permette che le applicazioni di elaborazione video falliscano o accumulino ritardo, recuperando senza alcuno sforzo aggiuntivo. Tramite `GetMedia`, le applicazioni sono in grado di elaborare i dati archiviati nel datastore e man mano che l'applicazione si aggiorna, `GetMedia` continua ad aggiornare i dati multimediali in tempo reale al loro arrivo.

- **GetMediaFromFragmentList** (e `ListFragments`): le applicazioni per l'elaborazione in batch sono considerate consumer offline. I consumatori offline potrebbero scegliere di recuperare in modo esplicito particolari frammenti di file multimediali o intervalli di video combinando le API `ListFragments` e `GetMediaFromFragmentList`. `ListFragments` e `GetMediaFromFragmentList` consentono a un'applicazione di identificare i segmenti di video

per un determinato intervallo di tempo o intervallo di frammenti, quindi di recuperare i frammenti in modo sequenziale o in parallelo per l'elaborazione. Questo approccio è idoneo per suite di applicazioni MapReduce, che devono elaborare rapidamente grandi quantità di dati in parallelo.

Ad esempio, supponiamo che un consumer voglia elaborare i frammenti di video di un intero giorno. Il consumer procederebbe nel modo seguente:

1. Ottenere un elenco di frammenti chiamando l'API `ListFragments` e specificando un intervallo di tempo per selezionare la raccolta di frammenti desiderata.

L'API restituisce i metadati da tutti i frammenti nell'intervallo di tempo specificato. I metadati forniscono informazioni come il numero di frammenti, i timestamp sul lato produttore e sul lato server e così via.

2. Prendere l'elenco dei metadati dei frammenti e recuperare i frammenti, in qualsiasi ordine. Ad esempio, per elaborare tutti i frammenti della giornata, il consumatore potrebbe scegliere di dividere l'elenco in sottoliste e fare in modo che i worker (ad esempio, più istanze di Amazon EC2) recuperino i frammenti in parallelo `GetMediaFromFragmentList` utilizzando e li elaborino in parallelo.

Il seguente diagramma mostra il flusso di dati per frammenti e blocchi durante queste chiamate API.

Quando un producer invia una richiesta `PutMedia`, invia metadati multimediali nel payload nonché una sequenza di frammenti di dati multimediali. Dopo aver ricevuto i dati, Kinesis Video Streams archivia i dati multimediali in entrata come blocchi Kinesis Video Streams. Ogni blocco è costituito dagli elementi seguenti:

- Una copia dei metadati multimediali
- Un frammento
- Metadati specifici di Kinesis Video Streams; ad esempio, il numero del frammento e i timestamp lato server e lato produttore

Quando un consumatore richiede metadati multimediali, Kinesis Video Streams restituisce un flusso di blocchi, a partire dal numero di frammento specificato nella richiesta.

Se abiliti la persistenza dei dati per lo stream, dopo aver ricevuto un frammento nello stream, Kinesis Video Streams salva anche una copia del frammento nell'archivio dati.

Modello di rilevamento degli endpoint

API REST di Control Plane

Per accedere alle API REST di [Kinesis Video Streams Control Plane](#), utilizza gli endpoint del servizio [Kinesis Video Streams](#).

API REST di Data Plane

Kinesis Video Streams è costruito [utilizzando un'](#)architettura cellulare per garantire migliori proprietà di scalabilità e isolamento del traffico. Poiché ogni stream è mappato su una cella specifica in una regione, l'applicazione deve utilizzare gli endpoint corretti specifici della cella a cui è stato mappato lo stream. Quando accedi alle API REST di Data Plane, dovrai gestire e mappare personalmente gli endpoint corretti. Questo processo, il modello di scoperta degli endpoint, è descritto di seguito:

1. Il pattern di rilevamento degli endpoint inizia con una chiamata a una delle `GetEndpoints` azioni. Queste azioni appartengono al Control Plane.
 1. Se stai recuperando gli endpoint per i [the section called “Amazon Kinesis Video Streams”](#) servizi [the section called “Amazon Kinesis Video Streams”](#) or, usa. [the section called “GetDataEndpoint”](#)
 2. Se stai recuperando gli endpoint per [the section called “Flesso di video Amazon Kinesis”, the section called “Amazon Kinesis Video Video Video Video Video Video Video Video WebRTC Video Video Video Video Video Video Video Video”](#) o [Kinesis](#) Video Signaling, usa. [the section called “GetSignalingChannelEndpoint”](#)
2. Memorizza nella cache e riutilizza l'endpoint.
3. Se l'endpoint memorizzato nella cache non funziona più, effettua una nuova chiamata `GetEndpoints` a per aggiornare l'endpoint.

Librerie Producer

Dopo aver creato uno stream video Kinesis, puoi iniziare a inviare dati allo stream. Nel codice dell'applicazione, puoi utilizzare queste librerie per estrarre dati dalle tue fonti multimediali e caricarli nel tuo flusso video Kinesis. Per ulteriori informazioni sulle librerie di producer disponibili, consulta [Librerie per produttori di Kinesis Video Streams](#).

Riproduzione di Kinesis Video Streams

È possibile visualizzare un flusso video Kinesis utilizzando i seguenti metodi:

- **GetMedia**— È possibile utilizzare l'GetMediaAPI per creare applicazioni personalizzate per l'elaborazione di Kinesis Video Streams. GetMedia è un'API in tempo reale con bassa latenza. Per creare un giocatore che lo utilizzi GetMedia, devi crearlo tu stesso. Per informazioni su come sviluppare un'applicazione che visualizza un flusso video Kinesis utilizzando GetMedia, consulta [Libreria parser del flusso](#)
- **HLS** — [HTTP Live Streaming \(HLS\)](#) è un protocollo di comunicazione di streaming multimediale basato su HTTP standard del settore. Puoi utilizzare HLS per visualizzare un flusso video Kinesis, per la riproduzione dal vivo o per visualizzare video archiviati.

È possibile usare HLS per la riproduzione in tempo reale. La latenza è in genere compresa tra 3 e 5 secondi, ma può essere compresa tra 1 e 10 secondi, a seconda del caso d'uso, del lettore e delle condizioni di rete. Puoi usare un lettore di terze parti (ad esempio [Video.js](#) o [Google Shaka Player](#)) per visualizzare il flusso video fornendo l'URL della sessione di streaming HLS, a livello di programmazione o manualmente. Puoi anche riprodurre video inserendo l'URL della sessione di streaming HLS nella barra degli indirizzi dei browser [Apple Safari](#) o [Microsoft Edge](#).

- **MPEG-DASH** — [Dynamic Adaptive Streaming over HTTP \(DASH\)](#), noto anche come MPEG-DASH, è un protocollo di streaming adattivo con bitrate che consente lo streaming di alta qualità di contenuti multimediali su Internet fornito da server Web HTTP convenzionali.

È possibile utilizzare MPEG-DASH per la riproduzione in tempo reale. La latenza è in genere compresa tra 3 e 5 secondi, ma può essere compresa tra 1 e 10 secondi, a seconda del caso d'uso, del lettore e delle condizioni di rete. È possibile utilizzare un lettore di terze parti (come [dash.js](#) o [Google Shaka Player](#)) per visualizzare lo streaming video fornendo l'URL della sessione di streaming MPEG-DASH, a livello di programmazione o manualmente.

- **GetClip**— È possibile utilizzare l'GetClipAPI per scaricare una clip (in un file MP4) contenente i file multimediali su richiesta archiviati dal flusso video specificato nell'intervallo di tempo specificato. Per ulteriori informazioni, consulta l'API Reference. [GetClip](#)

Argomenti

- [Requisiti per la riproduzione dei brani video](#)
- [Riproduzione video con HLS](#)
- [Riproduzione video con MPEG-DASH](#)

Requisiti per la riproduzione dei brani video

Amazon Kinesis Video Streams supporta contenuti multimediali codificati in più formati. Se lo streaming video di Kinesis utilizza un formato non supportato da una delle quattro API elencate di seguito, utilizza [GetMedia](#) o [GetMediaForFragmentList](#), poiché non hanno limitazioni relative al tipo di traccia.

Argomenti

- [GetClip requisiti](#)
- [Requisiti URL GetDash StreamingSession](#)
- [Requisiti URL di GetHLS StreamingSession](#)
- [GetImages requisiti](#)

GetClip requisiti

Per ulteriori informazioni sull'API, consulta [GetClip](#).

Descrizione del brano 1	ID del codec Track 1	Descrizione del brano 2	ID del codec Track 2
Video H.264	V_MPEG/ISO/AVC	N/D	N/D
Video H.264	V_MPEG/ISO/AVC	audio AAC	A_AAC
Video H.264	V_MPEG/ISO/AVC	Audio G.711 (solo A-Law)	A_MS/ACM
Video H.265	V_MPEGH/ISO/HEVC	N/D	N/D
Video H.265	V_MPEGH/ISO/HEVC	audio AAC	A_AAC

Important

I dati privati del codec (CPD) contenuti in ogni frammento contengono informazioni di inizializzazione specifiche del codec, come la frequenza dei fotogrammi, la risoluzione e il profilo di codifica, necessarie per decodificare correttamente il frammento. Le modifiche CPD

non sono supportate tra i frammenti di destinazione della clip risultante. Il CPD deve rimanere coerente su tutti i supporti interrogati, altrimenti verrà restituito un errore.

Important

Le modifiche alla traccia non sono supportate. Le tracce devono rimanere coerenti su tutti i file multimediali interrogati. Viene restituito un errore se i frammenti dello stream passano da un formato contenente solo video ad avere sia audio che video, o se una traccia audio AAC viene sostituita da una traccia audio A-Law.

Requisiti URL GetDash StreamingSession

Per ulteriori informazioni sull'API, consulta [GetDASHStreamingSessionURL](#).

Descrizione del brano 1	ID del codec Track 1	Descrizione del brano 2	ID del codec Track 2
Video H.264	V_MPEG/ISO/AVC	N/D	N/D
Video H.264	V_MPEG/ISO/AVC	audio AAC	A_AAC
Video H.264	V_MPEG/ISO/AVC	Audio G.711 (solo A-Law)	A_MS/ACM
Video H.264	V_MPEG/ISO/AVC	Audio G.711 (solo U-Law)	A_MS/ACM
audio AAC	A_AAC	N/D	N/D
Video H.265	V_MPEGH/ISO/HEVC	N/D	N/D
Video H.265	V_MPEGH/ISO/HEVC	audio AAC	A_AAC

⚠ Important

I dati privati del codec (CPD) contenuti in ogni frammento contengono informazioni di inizializzazione specifiche del codec, come la frequenza dei fotogrammi, la risoluzione e il profilo di codifica, necessarie per decodificare correttamente il frammento. Le modifiche CPD non sono supportate durante una sessione di streaming. Il CPD deve rimanere coerente attraverso i media interrogati.

⚠ Important

Le modifiche alla traccia non sono supportate. Le tracce devono rimanere coerenti su tutti i file multimediali interrogati. Lo streaming fallirà se i frammenti dello stream passano da un formato contenente solo video ad avere sia audio che video, o se una traccia audio AAC viene sostituita da una traccia audio A-Law.

Requisiti URL di GetHLS StreamingSession

Per ulteriori informazioni sull'API, consulta [GetHLSStreamingSessionURL](#).

HLS Mp4

Descrizione del brano 1	ID del codec Track 1	Descrizione del brano 2	ID del codec Track 2
Video H.264	V_MPEG/ISO/AVC	N/D	N/D
Video H.264	V_MPEG/ISO/AVC	audio AAC	A_AAC
Audio AAC	A_AAC	N/D	N/D
Video H.265	V_MPEGH/ISO/HEVC	N/D	N/D
Video H.265	V_MPEGH/ISO/HEVC	audio AAC	A_AAC

HLS TS

Descrizione del brano 1	ID del codec Track 1	Descrizione del brano 2	ID del codec Track 2
Video H.264	V_MPEG/ISO/AVC	N/D	N/D
Video H.264	V_MPEG/ISO/AVC	audio AAC	A_AAC
Audio AAC	A_AAC	N/D	N/D

Note

I dati privati del codec (CPD) contenuti in ogni frammento contengono informazioni di inizializzazione specifiche del codec, come la frequenza dei fotogrammi, la risoluzione e il profilo di codifica, necessarie per decodificare correttamente il frammento. Sia per TS che per MP4, le modifiche CPD sono supportate durante una sessione di streaming. Pertanto, i frammenti di una sessione possono contenere informazioni diverse nel CPD senza interrompere la riproduzione. Per ogni sessione di streaming, sono consentite solo 500 modifiche CPD.

Important

Le modifiche alla traccia non sono supportate. Le tracce devono rimanere coerenti su tutti i file multimediali interrogati. Lo streaming fallirà se i frammenti dello stream passano da un formato contenente solo video ad avere sia audio che video, o se una traccia audio AAC viene sostituita da una traccia audio A-Law.

GetImages requisiti

Per ulteriori informazioni sull'API, consulta [GetImages](#).

Note

Il GetImages file multimediale deve contenere una traccia video nella traccia 1.

Riproduzione video con HLS

[HTTP Live Streaming \(HLS\)](#) è un protocollo di comunicazione di streaming multimediale basato su HTTP standard del settore. Puoi utilizzare HLS per visualizzare un flusso video Kinesis, per la riproduzione dal vivo o per visualizzare video archiviati.

È possibile usare HLS per la riproduzione in tempo reale. La latenza è in genere compresa tra 3 e 5 secondi, ma può essere compresa tra 1 e 10 secondi, a seconda del caso d'uso, del lettore e delle condizioni di rete. Puoi usare un lettore di terze parti (ad esempio [Video.js](#) o [Google Shaka Player](#)) per visualizzare il flusso video fornendo l'URL della sessione di streaming HLS, a livello di programmazione o manualmente. Puoi anche riprodurre video inserendo l'URL della sessione di streaming HLS nella barra degli indirizzi dei browser [Apple Safari](#) o [Microsoft Edge](#).

[Per visualizzare uno streaming video Kinesis utilizzando HLS, crea innanzitutto una sessione di streaming utilizzando l'URL GetHLS. StreamingSession](#) Questa operazione restituisce un URL (contenente un token di sessione) per l'accesso alla sessione HLS. Puoi quindi usare l'URL in un lettore multimediale o in un'applicazione autonoma per visualizzare il flusso.

Important

Non tutti i file multimediali inviati a Kinesis Video Streams possono essere riprodotti tramite HLS. Vedi [the section called “GetHLSStreamingSessionURL”](#) per i requisiti di caricamento specifici.

Argomenti

- [Usa AWS CLI per recuperare l'URL di una sessione di streaming HLS](#)
- [Esempio: usa HLS in HTML e JavaScript](#)
- [Risoluzione dei problemi relativi a HLS](#)

Usa AWS CLI per recuperare l'URL di una sessione di streaming HLS

Segui la procedura riportata di seguito per utilizzare AWS CLI per generare un URL di sessione di streaming HLS per un flusso video Kinesis.

Per le istruzioni di installazione, consulta la Guida per l'[AWS Command Line Interface utente](#). Dopo l'installazione, [configura AWS CLI](#) con credenziali e regione.

In alternativa, apri il AWS CloudShell terminale, su cui è AWS CLI installato e configurato. Per ulteriori informazioni, [consulta la Guida per l'AWS CloudShell utente](#).

Recupera l'endpoint URL HLS per il tuo streaming video Kinesis.

1. Digita quanto segue nel terminale:

```
aws kinesismedia get-data-endpoint \
  --api-name GET_HLS_STREAMING_SESSION_URL \
  --stream-name YourStreamName
```

Riceverai una risposta simile alla seguente:

```
{
  "DataEndpoint": "https://b-1234abcd.kinesisvideo.aws-region.amazonaws.com"
}
```

2. Effettua la richiesta dell'URL della sessione di streaming HLS all'endpoint restituito.

Live

Per la riproduzione dal vivo, la playlist multimediale HLS viene continuamente aggiornata con i contenuti multimediali più recenti non appena disponibili. Quando riproduci questo tipo di sessione in un lettore multimediale, l'interfaccia utente visualizza in genere una notifica "live", senza alcun controllo di scorrimento per scegliere la posizione nella finestra di riproduzione da visualizzare.

Assicurati di caricare file multimediali su questo stream quando esegui questo comando.

```
aws kinesismedia get-hls-streaming-session-url \
  --endpoint-url https://b-1234abcd.kinesisvideo.aws-region.amazonaws.com \
  --stream-name YourStreamName \
  --playback-mode LIVE
```

Live replay

Per il replay dal vivo, la riproduzione inizia a partire da un'ora di inizio specificata. La playlist multimediale HLS viene inoltre continuamente aggiornata con i contenuti multimediali più recenti non appena disponibili. La sessione continuerà a includere i nuovi file multimediali acquisiti fino alla scadenza della sessione o fino all'ora di fine specificata, a seconda di

quale evento si verifichi per primo. Questa modalità è utile per poter avviare la riproduzione dal momento in cui viene rilevato un evento e continuare lo streaming live di contenuti multimediali che non sono ancora stati acquisiti al momento della creazione della sessione.

Determina un timestamp di inizio.

Per questo esempio, utilizziamo il formato dell'ora Unix Epoch in secondi. Fate riferimento alla sezione [Timestamp](#) della Guida per l' AWS Command Line Interface utente per ulteriori informazioni sulla formattazione dei timestamp.

Vedi [UnixTime.org](#) per uno strumento di conversione.

- 1708471800 è uguale al 20 febbraio 2024 15:30:00 GMT - 08:00

In questo esempio, non specifichiamo un timestamp di fine, il che significa che la sessione continuerà a includere i nuovi file multimediali importati fino alla scadenza della sessione.

[Richiama l'GetHLSStreamingSessionURLAPI con la modalità di LIVE_REPLAY riproduzione e un selettore di frammenti HLS specificato.](#)

```
aws kinesis-video-archived-media get-hls-streaming-session-url \  
  --endpoint-url https://b-1234abcd.kinesisvideo.aws-region.amazonaws.com \  
  --stream-name YourStreamName \  
  --playback-mode LIVE_REPLAY \  
  --hls-fragment-selector \  
  
"FragmentSelectorType=SERVER_TIMESTAMP, TimestampRange={StartTimestamp=1708471800}"
```

On-demand

Per la riproduzione su richiesta, la playlist multimediale HLS contiene i file multimediali specificati dal selettore di frammenti HLS. Quando questo tipo di sessione viene riprodotta in un lettore multimediale, l'interfaccia utente in genere visualizza un controllo a scorrimento per scegliere la posizione da visualizzare nella finestra di riproduzione.

Per creare un URL per una determinata sezione dello stream, stabilite innanzitutto i timestamp di inizio e fine.

Per questo esempio, utilizziamo il formato dell'ora Unix Epoch in secondi. Fate riferimento alla sezione [Timestamp](#) della Guida per l' AWS Command Line Interface utente per ulteriori informazioni sulla formattazione dei timestamp.

Vedi [UnixTime.org](https://unixtime.org) per uno strumento di conversione.

- 1708471800 è uguale al 20 febbraio 2024 15:30:00 GMT - 08:00
- 1708471860 è uguale al 20 febbraio 2024 15:31:00 GMT- 08:00

[Richiama l'GetHLSStreamingSessionURLAPI con ON_DEMAND la modalità di riproduzione e un selettore di frammenti HLS specificato.](#)

```
aws kinesis-video-archived-media get-hls-streaming-session-url \  
  --endpoint-url https://b-1234abcd.kinesisvideo.aws-region.amazonaws.com \  
  --stream-name YourStreamName \  
  --playback-mode ON_DEMAND \  
  --hls-fragment-selector \  
  
"FragmentSelectorType=SERVER_TIMESTAMP, TimestampRange={StartTimeStamp=1708471800, EndTimeStamp=1708471860}"
```

Note

I timestamp devono trovarsi entro 24 ore l'uno dall'altro, come indicato nella documentazione. [the section called "HLSTimestampRange"](#)

Riceverai una risposta simile alla seguente:

```
{  
  "HLSStreamingSessionURL": "https://b-1234abcd.kinesisvideo.aws-region.amazonaws.com/hls/v1/getHLSMasterPlaylist.m3u8?SessionToken=CiAz...DkRE6M~"  
}
```

⚠ Important

Non condividere o archiviare questo token in modo che un'entità non autorizzata possa accedervi. Il token fornisce l'accesso al contenuto dello stream. Proteggi il token con le stesse misure che utilizzeresti con AWS le tue credenziali.

Puoi utilizzare questo URL e qualsiasi lettore HLS per visualizzare lo stream HLS.

Ad esempio, usa VLC media player.

Puoi anche riprodurre HLS Stream inserendo l'URL della sessione di streaming HLS nella barra degli indirizzi del browser Apple Safari o Microsoft Edge.

Esempio: usa HLS in HTML e JavaScript

L'esempio seguente mostra come utilizzare l' AWS SDK per la versione JavaScript 2 per recuperare una sessione di streaming HLS per un flusso video Kinesis e riprodurla in una pagina Web.

L'esempio mostra come riprodurre video con i lettori seguenti:

- [Video.js](#)
- [Google Shaka Player](#)
- [hls.js](#)

[Visualizza il codice di esempio completo e la pagina web ospitata in.](#) GitHub

Argomenti illustrati nel codice:

- [Importa l' AWS SDK JavaScript per i browser](#)
- [Configurazione del client Kinesis Video Streams](#)
- [Recupera l'endpoint per la riproduzione HLS](#)
- [Configurare il client multimediale archiviato Kinesis Video Streams](#)
- [Recupera l'URL della sessione di streaming HLS](#)
- [Visualizza lo stream HLS sulla pagina web](#)

Importa l' AWS SDK JavaScript per i browser

Nella pagina Web, includi il seguente tag di script per importare l' AWS SDK per JavaScript v2 nel progetto.

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/aws-sdk/2.490.0/aws-sdk.min.js"></script>
```

Per ulteriori informazioni, consulta l'[AWS SDK per la documentazione. JavaScript](#)

Configurazione del client Kinesis Video Streams

Per accedere ai video in streaming con HLS, devi prima creare e configurare il client Kinesis Video Streams. Vedi [Impostazione delle credenziali in un browser Web](#) per altri metodi di autenticazione.

```
const clientConfig = {
  accessKeyId: 'YourAccessKey',
  secretAccessKey: 'YourSecretKey',
  region: 'us-west-2'
};
const kinesisVideoClient = new AWS.KinesisVideo(clientConfig);
```

L'applicazione recupera i valori necessari dalle caselle di input nella pagina HTML.

Recupera l'endpoint per la riproduzione HLS

Usa il client Kinesis Video Streams per [the section called "GetDataEndpoint"](#) richiamare l'API per recuperare l'endpoint.

```
const getDataEndpointOptions = {
  StreamName: 'YourStreamName',
  APIName: 'GET_HLS_STREAMING_SESSION_URL'
};
const getDataEndpointResponse = await kinesisVideoClient
  .getDataEndpoint(getDataEndpointOptions)
  .promise();
const hlsDataEndpoint = getDataEndpointResponse.DataEndpoint;
```

Questo codice memorizza l'endpoint nella variabile. `hlsDataEndpoint`

Configurare il client multimediale archiviato Kinesis Video Streams

Nella configurazione del client per il client multimediale archiviato Kinesis Video Streams, specifica l'endpoint ottenuto nel passaggio precedente.

```
const archivedMediaClientConfig = {
  accessKeyId: 'YourAccessKey',
  secretAccessKey: 'YourSecretKey',
  region: 'us-west-2',
  endpoint: hlsDataEndpoint
};
const kinesisVideoArchivedMediaClient = new
  AWS.KinesisVideoArchivedMedia(archivedMediaClientConfig);
```

Recupera l'URL della sessione di streaming HLS

Utilizza il client multimediale archiviato Kinesis Video Streams per [the section called "GetHLSStreamingSessionURL"](#) richiamare l'API per recuperare l'URL di riproduzione HLS.

```
const getHLSStreamingSessionURLOptions = {
  StreamName: 'YourStreamName',
  PlaybackMode: 'LIVE'
};
const getHLSStreamingSessionURLResponse = await kinesisVideoArchivedMediaClient
  .getHLSStreamingSessionURL(getHLSStreamingSessionURLOptions)
  .promise();
const hlsUrl = getHLSStreamingSessionURLResponse.HLSStreamingSessionURL;
```

Visualizza lo stream HLS sulla pagina web

Una volta ottenuto l'URL della sessione di streaming HLS, forniscilo al lettore video. Il metodo per fornire l'URL al lettore video dipende dal lettore usato.

Video.js

Effettua le seguenti operazioni per importare [Video.js](#) e le sue classi CSS nello script del nostro browser:

```
<link rel="stylesheet" href="https://vjs.zencdn.net/6.6.3/video-js.css">
<script src="https://vjs.zencdn.net/6.6.3/video.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/videojs-contrib-hls/5.14.1/
videojs-contrib-hls.js"></script>
```

Create un elemento video HTML per visualizzare il video:

```
<video id="videojs" class="player video-js vjs-default-skin" controls autoplay></video>
```

Imposta l'URL HLS come sorgente dell'elemento video HTML:

```
const playerElement = document.getElementById('videojs');
const player = videojs(playerElement);
player.src({
  src: hlsUrl,
  type: 'application/x-mpegURL'
});
player.play();
```

Shaka

Fai quanto segue per importare il [lettore Google Shaka nello script](#) del nostro browser:

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/shaka-player/2.4.1/shaka-player.compiled.js"></script>
```

Crea un elemento video HTML per visualizzare il video:

```
<video id="shaka" class="player" controls autoplay></video>
```

Creare un player Shaka specificando l'elemento video e chiamate il metodo load:

```
const playerElement = document.getElementById('shaka');
const player = new shaka.Player(playerElement);
player.load(hlsUrl);
```

hls.js

Effettua quanto segue per importare [hls.js](#) nello script del nostro browser:

```
<script src="https://cdn.jsdelivr.net/npm/hls.js@latest"></script>
```

Crea un elemento video HTML per visualizzare il video:

```
<video id="hlsjs" class="player" controls autoplay></video>
```

Crea un player hls.js, assegnagli l'URL HLS e digli di giocare:

```
const playerElement = document.getElementById('hlsjs');
const player = new Hls();
player.loadSource(hlsUrl);
player.attachMedia(playerElement);
player.on(Hls.Events.MANIFEST_PARSED, function() {
    video.play();
});
```

Risoluzione dei problemi relativi a HLS

Questa sezione descrive i problemi che potresti riscontrare durante l'utilizzo di HTTP Live Streaming (HLS) con Kinesis Video Streams.

Problemi

- [Il recupero dell'URL della sessione di streaming HLS riesce, ma si verifica un problema di riproduzione nel lettore video](#)
- [Latenza troppo elevata tra il producer e il lettore](#)

Il recupero dell'URL della sessione di streaming HLS riesce, ma si verifica un problema di riproduzione nel lettore video

Questa situazione si verifica quando è possibile recuperare correttamente l'URL di una sessione di streaming HLS usando `GetHLSStreamingSessionURL`, ma non è possibile riprodurre il video fornendo l'URL a un lettore video.

Per risolvere questo problema, prova le soluzioni seguenti:

- Determina se il flusso video viene riprodotto nella console Kinesis Video Streams. Esamina gli errori visualizzati nella console.
- Se la durata del frammento è meno di un secondo, aumentala a un secondo. Se la durata del frammento è troppo breve, il servizio potrebbe rallentare il lettore perché effettua richieste di frammenti video troppo frequentemente.
- Verificare che ogni URL di una sessione di streaming HLS venga usato da un solo lettore. Se più di un lettore usa un singolo URL di una sessione di streaming HLS, il servizio potrebbe ricevere troppe richieste e limitarle.

- Verifica che il lettore supporti tutte le opzioni specificate per la sessione di streaming HLS. Prova combinazioni diverse di valori per i parametri seguenti:
 - `ContainerFormat`
 - `PlaybackMode`
 - `FragmentSelectorType`
 - `DiscontinuityMode`
 - `MaxMediaPlaylistFragmentResults`

Alcuni lettori multimediali (ad esempio HTML5 e lettori mobile) in genere supportano solo HLS con il formato container fMP4. Altri lettori multimediali (come Flash e lettori personalizzati) potrebbero supportare HLS solo con il formato contenitore MPEG TS. Ti consigliamo di provare il `ContainerFormat` parametro per iniziare la risoluzione dei problemi.

- Verificare che ogni frammento abbia un numero consistente di tracce. Verifica che i frammenti dello stream non contengano solo una traccia video e audio e non contengano solo una traccia video. Verifica inoltre che le impostazioni del codificatore (risoluzione e frequenza fotogrammi) non cambino tra i frammenti di ogni traccia.

Latenza troppo elevata tra il producer e il lettore

Questa situazione si verifica quando vi è una latenza troppo elevata dal momento dell'acquisizione del video alla sua riproduzione nel lettore.

Il video viene riprodotto tramite HLS frammento per frammento. Pertanto, la latenza non può essere inferiore alla durata dei frammenti. La latenza include anche il tempo necessario per il buffering e il trasferimento dei dati. Se la soluzione richiede una latenza inferiore a un secondo, usa l'API `GetMedia`.

È possibile modificare i parametri seguenti per ridurre la latenza complessiva, ma la modifica di questi parametri potrebbe comportare anche la riduzione della qualità video o l'aumento della frequenza di ripetizione del buffering.

- **Durata del frammento:** la durata del frammento è la quantità di video tra le divisioni del flusso, controllata dalla frequenza dei fotogrammi chiave generati dal codificatore video. Il valore consigliato è di un secondo. Una durata inferiore dei frammenti significa meno tempo di attesa per il completamento del frammento prima della trasmissione dei dati video al servizio. I frammenti più brevi vengono anche elaborati più velocemente dal servizio. Tuttavia, se la durata di frammento è troppo breve, aumentano le probabilità che il lettore esaurisca i contenuti e debba arrestarsi ed

eseguire il buffering di altri contenuti. Se la durata del frammento è meno di 500 millisecondi, il producer potrebbe creare troppe richieste, che potrebbero pertanto essere limitate dal servizio.

- **Bitrate:** uno streaming video con un bitrate inferiore richiede meno tempo per la lettura, la scrittura e la trasmissione. Tuttavia, un flusso video con una velocità in bit inferiore ha in genere una qualità video inferiore.
- **Numero di frammenti nelle playlist multimediali:** un lettore sensibile alla latenza dovrebbe caricare solo i frammenti più recenti di una playlist multimediale. La maggior parte dei giocatori inizia invece dal primo frammento. Riducendo il numero di frammenti nella playlist, riduci la distanza temporale tra il frammento precedente e quello nuovo. Con una playlist di dimensioni inferiori, è possibile che un frammento venga saltato durante la riproduzione se c'è un ritardo nell'aggiunta di nuovi frammenti alla playlist o se c'è un ritardo nell'ottenere una playlist aggiornata. Ti consigliamo di usare 3-5 frammenti e di utilizzare un player configurato per caricare solo i frammenti più recenti da una playlist.
- **Dimensione del buffer del lettore:** la maggior parte dei lettori video ha una durata minima del buffer configurabile, di solito con un valore predefinito di 10 secondi. Per ottenere la latenza più bassa, è possibile impostare questo valore su 0 secondi. Tuttavia, così facendo, il lettore esegue il rebuffer se c'è un ritardo nella produzione di frammenti, perché non disporrà di un buffer per assorbire il ritardo.
- **Player «catch up»:** i lettori video in genere non riprendono automaticamente la riproduzione fino alla parte anteriore del buffer video se il buffer si riempie, come quando un frammento ritardato causa la riproduzione di un backlog di frammenti. Un lettore personalizzato consente di evitare questa situazione, eliminando i fotogrammi o aumentando la velocità di riproduzione (ad esempio a 1,1x) per raggiungere la parte iniziale del buffer. Ciò causa una riproduzione irregolare o un aumento della velocità mentre il lettore recupera e la ripetizione del buffering potrebbe avvenire con maggiore frequenza se le dimensioni del buffer sono contenute.

Riproduzione video con MPEG-DASH

[Per visualizzare uno streaming video Kinesis utilizzando MPEG-DASH, devi prima creare una sessione di streaming utilizzando GetDash URL. `StreamingSession`](#) Questa operazione restituisce un URL (contenente un token di sessione) per l'accesso alla sessione MPEG-DASH. Puoi quindi usare l'URL in un lettore multimediale o in un'applicazione autonoma per visualizzare il flusso.

Uno streaming video di Amazon Kinesis presenta i seguenti requisiti per la fornitura di video tramite MPEG-DASH:

- Per i requisiti di riproduzione dei brani video in streaming, consulta [the section called “URL GetDash StreamingSession”](#)
- La conservazione dei dati deve essere superiore a 0.
- La traccia video di ogni frammento deve contenere i dati codec privati in Advanced Video Coding (AVC) per il formato H.264 e HEVC per il formato H.265. Per ulteriori informazioni, consulta [Specifica MPEG-4 ISO/IEC 14496-15](#). Per informazioni sull'adattamento dei dati di flusso a un dato formato, vedi i [contrassegni di adattamento NAL](#).
- La traccia audio (se presente) di ogni frammento deve contenere dati codec privati in formato AAC ([Specifica AAC ISO/IEC 13818-7](#)) o in formato [MS Wave](#).

Esempio: utilizzo di MPEG-DASH in HTML e JavaScript

L'esempio seguente mostra come recuperare una sessione di streaming MPEG-DASH per un flusso video Kinesis e riprodurla in una pagina Web. L'esempio mostra come riprodurre video con i lettori seguenti:

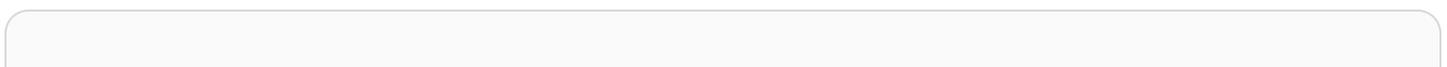
- [Google Shaka Player](#)
- [dash.js](#)

Argomenti

- [Configurazione del client Kinesis Video Streams per la riproduzione MPEG-DASH](#)
- [Recupera l'endpoint dei contenuti archiviati di Kinesis Video Streams per la riproduzione in formato MPEG-DASH](#)
- [Recuperare l'URL della sessione di streaming MPEG-DASH](#)
- [Visualizza il video di streaming con la riproduzione MPEG-DASH](#)
- [Esempio completo](#)

Configurazione del client Kinesis Video Streams per la riproduzione MPEG-DASH

Per accedere ai video in streaming con MPEG-DASH, devi prima creare e configurare il client Kinesis Video Streams (per recuperare l'endpoint del servizio) e il client multimediale archiviato (per recuperare la sessione di streaming MPEG-DASH). L'applicazione recupera i valori necessari dalle caselle di input nella pagina HTML.



```

var streamName = $('#streamName').val();

// Step 1: Configure SDK Clients
var options = {
  accessKeyId: $('#accessKeyId').val(),
  secretAccessKey: $('#secretAccessKey').val(),
  sessionToken: $('#sessionToken').val() || undefined,
  region: $('#region').val(),
  endpoint: $('#endpoint').val() || undefined
}
var kinesisVideo = new AWS.KinesisVideo(options);
var kinesisVideoArchivedContent = new AWS.KinesisVideoArchivedMedia(options);

```

Recupera l'endpoint dei contenuti archiviati di Kinesis Video Streams per la riproduzione in formato MPEG-DASH

Dopo l'avvio dei client, recupera l'endpoint dei contenuti archiviati di Kinesis Video Streams in modo da poter recuperare l'URL della sessione di streaming MPEG-DASH come segue:

```

// Step 2: Get a data endpoint for the stream
console.log('Fetching data endpoint');
kinesisVideo.getDataEndpoint({
  StreamName: streamName,
  APIName: "GET_DASH_STREAMING_SESSION_URL"
}, function(err, response) {
  if (err) { return console.error(err); }
  console.log('Data endpoint: ' + response.DataEndpoint);
  kinesisVideoArchivedContent.endpoint = new AWS.Endpoint(response.DataEndpoint);
}

```

Recuperare l'URL della sessione di streaming MPEG-DASH

Quando disponi dell'endpoint per i contenuti archiviati, richiama l'API URL

[StreamingSessionGetDASH](#) per recuperare l'URL della sessione di streaming MPEG-DASH nel modo seguente:

```

// Step 3: Get a Streaming Session URL
var consoleInfo = 'Fetching ' + protocol + ' Streaming Session URL';
console.log(consoleInfo);

if (protocol === 'DASH') {

```

```

kinesisVideoArchivedContent.getDASHStreamingSessionURL({
  StreamName: streamName,
  PlaybackMode: $('#playbackMode').val(),
  DASHFragmentSelector: {
    FragmentSelectorType: $('#fragmentSelectorType').val(),
    TimestampRange: $('#playbackMode').val() === "LIVE" ? undefined : {
      StartTimestamp: new Date($('#startTimestamp').val()),
      EndTimestamp: new Date($('#endTimestamp').val())
    }
  },
  DisplayFragmentTimestamp: $('#displayFragmentTimestamp').val(),
  DisplayFragmentNumber: $('#displayFragmentNumber').val(),
  MaxManifestFragmentResults: parseInt($('#maxResults').val()),
  Expires: parseInt($('#expires').val())
}, function(err, response) {
  if (err) { return console.error(err); }
  console.log('DASH Streaming Session URL: ' + response.DASHStreamingSessionURL);
});

```

Visualizza il video di streaming con la riproduzione MPEG-DASH

Una volta ottenuto l'URL della sessione di streaming MPEG-DASH, forniscilo al lettore video. Il metodo per fornire l'URL al lettore video dipende dal lettore usato.

L'esempio di codice seguente illustra come fornire l'URL della sessione di streaming a un lettore [Google Shaka Player](#):

```

// Step 4: Give the URL to the video player.

//Shaka Player elements
<video id="shaka" class="player" controls autoplay></video>
<script src="https://cdnjs.cloudflare.com/ajax/libs/shaka-player/2.4.1/shaka-
player.compiled.js">
</script>
...

var playerName = $('#player').val();

if (playerName === 'Shaka Player') {
  var playerElement = $('#shaka');
  playerElement.show();
}

```

```
var player = new shaka.Player(playerElement[0]);
console.log('Created Shaka Player');

player.load(response.DASHStreamingSessionURL).then(function() {
  console.log('Starting playback');
});
console.log('Set player source');
}
```

L'esempio di codice seguente illustra come fornire l'URL della sessione di streaming a un lettore [dash.js](#):

```
<!-- dash.js Player elements -->
<video id="dashjs" class="player" controls autoplay=""></video>
<script src="https://cdn.dashjs.org/latest/dash.all.min.js"></script>

...

var playerElement = $('#dashjs');
playerElement.show();

var player = dashjs.MediaPlayer().create();
console.log('Created DASH.js Player');

player.initialize(document.querySelector('#dashjs'), response.DASHStreamingSessionURL,
  true);
console.log('Starting playback');
console.log('Set player source');
}
```

Esempio completo

È possibile [scaricare](#) o visualizzare il codice di esempio completato su GitHub

Utilizzo dei metadati di streaming con Kinesis Video Streams

Puoi utilizzare l'SDK Amazon Kinesis Video Streams Producer per incorporare i metadati a livello di singolo frammento in un flusso video Kinesis. I metadati in Kinesis Video Streams sono una coppia chiave-valore mutabile. Puoi utilizzarla per descrivere il contenuto del frammento, incorporare le letture dei sensori associate che devono essere trasferite insieme al frammento effettivo o soddisfare altre esigenze personalizzate. I metadati sono disponibili come parte delle operazioni delle API

[the section called “GetMedia”](#) o [the section called “GetMediaForFragmentList”](#). Viene archiviato insieme ai frammenti per l'intera durata del periodo di conservazione dello stream. Le applicazioni che consumano possono leggere, elaborare e reagire in base ai metadati utilizzando il [Libreria di analisi dei flussi video Kinesis](#)

Esistono due modalità in cui i metadati possono essere incorporati con i frammenti in un flusso:

- Non persistente: puoi applicare i metadati una tantum o ad hoc ai frammenti di un flusso, in base a criteri aziendali specifici che si sono verificati. Un esempio è una videocamera intelligente che rileva il movimento e aggiunge metadati ai frammenti corrispondenti che contengono il movimento prima di inviarli al relativo flusso video Kinesis. Puoi applicare i metadati al frammento nel formato seguente: `Motion = true`.
- Persistente: puoi aggiungere metadati a frammenti successivi e consecutivi in un flusso in base a un'esigenza continua. Un esempio è una videocamera intelligente che invia le coordinate correnti di latitudine e longitudine associate a tutti i frammenti che invia al flusso video Kinesis. Puoi applicare i metadati a tutti i frammenti nel formato seguente: `Lat = 47.608013N , Long = -122.335167W`.

Puoi applicare i metadati in entrambe queste modalità allo stesso frammento contemporaneamente in base alle esigenze dell'applicazione. I metadati incorporati potrebbero includere oggetti rilevati, attività tracciate, coordinate GPS o altri dati personalizzati che vuoi associare ai frammenti nel flusso. I metadati sono codificati come coppie di stringhe chiave-valore.

Argomenti

- [Aggiungere metadati a un flusso video Kinesis](#)
- [Utilizzo dei metadati incorporati in un flusso video Kinesis](#)
- [Limiti dei metadati di streaming](#)

Aggiungere metadati a un flusso video Kinesis

I metadati aggiunti a uno stream video Kinesis sono modellati come tag MKV, che vengono implementati come coppie chiave-valore.

I metadati possono essere transitori, ad esempio per contrassegnare un evento all'interno del flusso, o persistenti, ad esempio per identificare frammenti in cui si verifica un determinato evento. Un elemento di metadati persistente rimane e viene applicato a ogni frammento consecutivo finché non viene annullato.

Note

Gli elementi metadati aggiunti utilizzando le [Producer Library](#) differiscono dalle API di tag a livello di flusso implementato con [the section called "TagStream"](#), [the section called "UntagStream"](#) e [the section called "ListTagsForStream"](#).

API per lo streaming di metadati

Puoi utilizzare le seguenti operazioni in Producer SDK per implementare lo streaming dei metadati.

PIC

```
PUBLIC_API STATUS putKinesisVideoFragmentMetadata(STREAM_HANDLE streamHandle,
    PCHAR name,
    PCHAR value,
    BOOL persistent);
```

C++ Producer SDK

```
/**
 * Appends a "tag" or metadata - a key/value string pair into the stream.
 */
bool putFragmentMetadata(const std::string& name, const std::string& value, bool
    persistent = true);
```

Java Producer SDK

Puoi utilizzare Java Producer SDK per aggiungere metadati a un utilizzando: `MediaSource` `MediaSourceSink.onCodecPrivateData`

```
void onFragmentMetadata(final @NonNull String metadataName, final @NonNull String
    metadataValue, final boolean persistent)
    throws KinesisVideoException;
```

Metadati persistenti e non persistenti

Per i metadati non persistenti, puoi aggiungere più elementi metadati con lo stesso nome. Producer SDK raccoglie gli elementi metadati nella coda finché non vengono anteposti al successivo

frammento. La coda dei metadati viene cancellata quando gli elementi metadati vengono applicati al flusso. Per ripetere i metadati, chiama di nuovo `putKinesisVideoFragmentMetadata` o `putFragmentMetadata`.

Per i metadati persistenti, Producer SDK raccoglie gli elementi metadati nella coda come per i metadati non persistenti. Tuttavia, gli elementi di metadati non vengono rimossi dalla coda quando vengono aggiunti al frammento successivo.

La chiamata `putKinesisVideoFragmentMetadata` o `putFragmentMetadata` con `persistent` impostato su `true` ha il seguente comportamento:

- La chiamata dell'API inserisce l'elemento metadati nella coda. I metadati vengono aggiunti come tag MKV per ogni frammento mentre l'elemento è in coda.
- Se si chiama l'API con lo stesso nome e un valore diverso come elemento metadati aggiunto in precedenza, l'elemento viene sovrascritto.
- Se si chiama l'API con un valore vuoto, l'elemento metadati viene rimosso (annullato) dalla coda dei metadati.

Utilizzo dei metadati incorporati in un flusso video Kinesis

Per utilizzare i metadati in un flusso video Kinesis, utilizza un'implementazione di: `MkvTagProcessor`

```
public interface MkvTagProcessor {
    default void process(MkvTag mkvTag, Optional<FragmentMetadata>
currentFragmentMetadata) {
        throw new NotImplementedException("Default
FragmentMetadataVisitor.MkvTagProcessor");
    }
    default void clear() {
        throw new NotImplementedException("Default
FragmentMetadataVisitor.MkvTagProcessor");
    }
}
```

Questa interfaccia si trova nella classe [FragmentMetadataVisitor](#) nella [Libreria di analisi dei flussi video Kinesis](#).

La classe `FragmentMetadataVisitor` contiene un'implementazione di `MkvTagProcessor`:

```
public static final class BasicMkvTagProcessor implements
FragmentMetadataVisitor.MkvTagProcessor {
    @Getter
    private List<MkvTag> tags = new ArrayList<>();

    @Override
    public void process(MkvTag mkvTag, Optional<FragmentMetadata>
currentFragmentMetadata) {
        tags.add(mkvTag);
    }

    @Override
    public void clear() {
        tags.clear();
    }
}
```

La classe `KinesisVideoRendererExample` contiene un esempio di come usare un `BasicMkvTagProcessor`. Nel seguente esempio, un `BasicMkvTagProcessor` viene aggiunto a `MediaProcessingArguments` di un'applicazione:

```
if (renderFragmentMetadata) {
    getMediaProcessingArguments =
KinesisVideoRendererExample.GetMediaProcessingArguments.create(
        Optional.of(new FragmentMetadataVisitor.BasicMkvTagProcessor()));
}
```

Il metodo `BasicMkvTagProcessor.process` viene chiamato quando arrivano i metadati del frammento. Puoi recuperare i metadati accumulati con `GetTags`. Per recuperare un singolo elemento di metadati, prima chiama `clear` per cancellare i metadati raccolti, quindi recupera nuovamente gli elementi di metadati.

Limiti dei metadati di streaming

[the section called “Quote di metadati per frammenti”](#) Per ulteriori informazioni sui limiti che si applicano all'aggiunta di metadati di streaming a un flusso video Kinesis, consulta

Modello di dati Kinesis Video Streams

[Producer Library](#) e [Libreria parser del flusso](#) inviano e ricevono dati video in un formato che supporta l'incorporamento di informazioni insieme ai dati video. Questo formato si basa sulla specifica Matroska (MKV).

Il [formato MKV](#) è una specifica aperta per i dati multimediali. Tutte le librerie e gli esempi di codice contenuti nella Amazon Kinesis Video Streams Developer Guide inviano o ricevono dati in formato MKV.

[Librerie per produttori di Kinesis Video Streams](#) Utilizza i Frame tipi `StreamDefinition` e per produrre intestazioni di stream MKV, intestazioni di frame e dati di frame.

Per informazioni sulle specifiche complete di MKV, consulta l'articolo sulle [specifiche Matroska](#).

Le seguenti sezioni descrivono i componenti di dati in formato MKV creati da [C++ Producer Library](#).

Argomenti

- [Elementi di intestazione dello stream](#)
- [Trasmetti i dati della traccia](#)
- [Elementi dell'intestazione del frame](#)
- [Dati del frame MKV](#)

Elementi di intestazione dello stream

I seguenti elementi di intestazione MKV vengono utilizzati da `StreamDefinition` (definiti in `StreamDefinition.h`).

Elemento	Description	Valori tipici
<code>stream_name</code>	Corrisponde al nome dello stream video Kinesis.	my-stream
<code>retention_period</code>	La durata, in ore, del flusso di dati viene mantenuta da Kinesis Video Streams. 0 Specificalo per uno stream che non conserva dati.	24

Elemento	Description	Valori tipici
tags	Una raccolta chiave-valore di dati utente. Questi dati vengono visualizzati nella AWS Management Console e possono essere letti dalle applicazioni client per filtrare o ottenere informazioni su un flusso.	
kms_key_id	Se presente, la AWS KMS chiave definita dall'utente viene utilizzata per crittografare i dati sullo stream. Se assenti, i dati vengono crittografati dalla chiave fornita da Kinesis (<code>aws/kinesis-video</code>).	01234567-89ab-cdef-0123-456789ab
streaming_type	Attualmente, l'unico tipo di streaming valido è <code>STREAMING_TYPE_REALTIME</code> .	<code>STREAMING_TYPE_REALTIME</code>
content_type	Il tipo di contenuto definito dall'utente. Per lo streaming di dati video da riprodurre nella console, il tipo di contenuto deve essere <code>video/h264</code> .	<code>video/h264</code>
max_latency	Questo valore non è attualmente utilizzato e dovrebbe essere impostato su 0.	0

Elemento	Description	Valori tipici
<code>fragment_duration</code>	La stima della durata dei tuoi frammenti, utile per l'ottimizzazione. La durata effettiva è determinata dai dati di streaming.	2
<code>timecode_scale</code>	<p>Indica la scala utilizzata dai timestamp dei frame. Il valore predefinito è 1 millisecondo. Se si specifica 0, si assegna anche il valore predefinito di 1 millisecondo. Questo valore può essere compreso tra 100 nanosecondi e 1 secondo.</p> <p>Per ulteriori informazioni, consultate la documentazione TimecodeScale di Matroska.</p>	
<code>key_frame_fragmentation</code>	Se <code>true</code> , il flusso avvia un nuovo cluster quando riceve un keyframe.	<code>true</code>
<code>frame_timecodes</code>	Se <code>true</code> , Kinesis Video Streams utilizza i valori del timestamp di presentazione (<code>pts</code>) e del timestamp di decodifica (<code>dts</code>) dei frame ricevuti. Se <code>false</code> , Kinesis Video Streams stampa i fotogrammi quando vengono ricevuti con valori temporali generati dal sistema.	<code>true</code>

Elemento	Description	Valori tipici
<code>absolute_fragment_time</code>	Se <code>true</code> , i timecode del cluster vengono interpretati come se utilizzassero tempo assoluto (ad esempio, dall'orologio di sistema del producer) . Se <code>false</code> , i timecode del cluster sono interpretati come relativi all'ora di inizio del flusso.	<code>true</code>
<code>fragment_acks</code>	Se <code>true</code> , le conferme (ACK) vengono inviate quando Kinesis Video Streams riceve i dati. Le ACK possono essere ricevute utilizzando le callback <code>KinesisVideoStreamFragmentAck</code> o <code>KinesisVideoStreamParseFragmentAck</code> .	<code>true</code>
<code>restart_on_error</code>	Indica se il flusso deve riprendere la trasmissione dopo la segnalazione di un errore di flusso.	<code>true</code>
<code>nal_adaptation_flags</code>	Indica se i dati privati del codec o l'adattamento NAL (Network Abstraction Layer) sono presenti nel contenuto . I flag validi includono <code>NAL_ADAPTATION_ANNEXB_NALS</code> e <code>NAL_ADAPTATION_ANNEXB_CPD_NALS</code> .	<code>NAL_ADAPTATION_ANNEXB_NALS</code>

Elemento	Description	Valori tipici
frame_rate	Una stima della frequenza di frame del contenuto. Tale valore viene utilizzato per l'ottimizzazione. La frequenza di frame effettiva è determinata dalla velocità dei dati in entrata. Se si specifica 0, si assegna il valore predefinito di 24.	24
avg_bandwidth_bps	Una stima della larghezza di banda dei contenuti, in Mbps. Tale valore viene utilizzato per l'ottimizzazione. La frequenza effettiva è determinata dalla larghezza di banda dei dati in entrata. Ad esempio, per un flusso video con risoluzione di 720 p in esecuzione a 25 FPS, puoi aspettarti una larghezza di banda media di 5 Mbps.	5
buffer_duration	La durata del buffering del contenuto sul producer. Se la latenza di rete è bassa, questo valore può essere ridotto. Se la latenza di rete è elevata, aumentando questo valore si evita che i frame vengano eliminati prima che possano essere inviati, a causa dell'incapacità dell'allocation di inserire i frame nel buffer più piccolo.	

Elemento	Description	Valori tipici
replay_duration	La quantità di tempo in cui il flusso di dati video viene «riavvolto» in caso di interruzione della connessione. Questo valore può essere zero se la perdita di frame a causa della perdita di connessione non è un problema. Il valore può essere aumentato se l'applicazione che utilizza è in grado di rimuovere i frame ridondanti. Questo valore deve essere inferiore alla durata del buffer, altrimenti viene utilizzato a la durata del buffer.	
connection_staleness	Il periodo durante il quale una connessione viene mantenuta quando non vengono ricevuti dati.	
codec_id	Il codec utilizzato per il contenuto. Per ulteriori informazioni, consulta CodecID nelle specifiche Matroska.	V_MPEG2
track_name	Il nome definito dall'utente della traccia.	my_track

Elemento	Description	Valori tipici
codecPrivateData	I dati forniti dall'encoder utilizzato per decodificare i dati dei frame, ad esempio la larghezza di banda e l'altezza in pixel degli stessi, un'informazione richiesta da numerosi consumer a valle. In C++ Producer Library , la matrice <code>gMkvTrackVideoBits</code> in <code>MkvStatics.cpp</code> include la larghezza e l'altezza in pixel per il frame.	
codecPrivateDataDimensioni	La dimensione dei dati nel parametro <code>codecPrivateData</code> .	
track_type	Il tipo di traccia per il flusso.	MKV_TRACK_INFO_TYPE_AUDIO or MKV_TRACK_INFO_TYPE_VIDEO
segment_uuid	uuid segmento definito dall'utente (16 byte).	
default_track_id	Numero univoco diverso da zero per la traccia.	1

Trasmetti i dati della traccia

I seguenti elementi della traccia MKV vengono utilizzati da `StreamDefinition` (definiti in `StreamDefinition.h`).

Elemento	Description	Valori tipici
track_name	Nome traccia definito dall'utente. Ad esempio, "audio" per la traccia audio.	audio
codec_id	Id codec della traccia. Ad esempio, "A_AAC" per la traccia audio.	A_AAC
cpd	I dati forniti dall'encoder utilizzato per decodificare i dati del frame. Queste informazioni possono includere larghezza e altezza del frame in pixel, necessarie per numerosi consumatori downstream. Nella libreria C++ Producer , l' gMkvTrackVideoBits array in MkvStatics .cpp include la larghezza e l'altezza dei pixel per la cornice.	
cpd_size	La dimensione dei dati nel parametro. codecPrivateData	
track_type	Il tipo di traccia. Ad esempio, è possibile utilizzare il valore di enum MKV_TRACK_INFO_TYPE_AUDIO per l'audio.	MKV_TRACK_INFO_TYPE_AUDIO

Elementi dell'intestazione del frame

I seguenti elementi di intestazione MKV vengono utilizzati da `Frame` (definiti nel pacchetto `KinesisVideoPic`, in `mkvgen/Include.h`):

- **Frame Index (Indice del frame):** un valore a incremento monotono.
- **Flags (Flag):** il tipo di frame. I valori validi includono i seguenti:
 - `FRAME_FLAGS_NONE`
 - `FRAME_FLAG_KEY_FRAME`: se `key_frame_fragmentation` è impostato su flusso, i key frame avviano un nuovo frammento.
 - `FRAME_FLAG_DISCARDABLE_FRAME`: indica al decoder che è possibile ignorare questo frame se la decodifica è lenta.
 - `FRAME_FLAG_INVISIBLE_FRAME`: la durata di questo blocco è 0.
- **Timestamp di decodifica:** il timestamp di quando questo frame è stato decodificato. Se i frame precedenti dipendono da questo frame per la decodifica, questo timestamp potrebbe essere precedente a quello dei frame precedenti. Questo valore è relativo all'inizio del frammento.
- **Timestamp di presentazione:** il timestamp di quando viene visualizzato questo frame. Questo valore è relativo all'inizio del frammento.
- **Duration (Durata):** la durata di riproduzione del frame.
- **Size (Dimensione):** la dimensione dei dati del frame in byte

Dati del frame MKV

I dati in `frame.frameData` potrebbero contenere solo i dati multimediali per il frame o ulteriori informazioni di intestazione annidate, a seconda dello schema di codifica utilizzato. Per essere visualizzati in AWS Management Console, i dati devono essere codificati nel codec [H.264](#), ma Kinesis Video Streams può ricevere flussi di dati con serializzazione temporale in qualsiasi formato.

Guida introduttiva ad Amazon Kinesis Video Streams

Questa sezione descrive come eseguire le seguenti attività in Amazon Kinesis Video Streams:

- Configura il tuo Account AWS e crea un amministratore, se non l'hai già fatto.
- Crea un flusso video Kinesis.
- Invia i dati allo streaming video Kinesis dalla videocamera e visualizza i contenuti multimediali nella console.

Se non conosci Amazon Kinesis Video Streams, ti consigliamo [Kinesis Video Streams: come funziona](#) di leggere prima.

Note

Se segui l'esempio di Guida introduttiva non dovrai sostenere alcun costo per il tuo Account AWS. Per i costi dei dati nella tua regione, consulta i prezzi di [Amazon Kinesis Video Streams](#).

Argomenti

- [Configurazione di un account](#)
- [Creare un flusso video Kinesis](#)
- [Inviare dati a un flusso video di Amazon Kinesis](#)
- [Consuma dati multimediali](#)

Configurazione di un account

Prima di utilizzare Amazon Kinesis Video Streams per la prima volta, completa le seguenti attività.

Argomenti

- [Registrati per un Account AWS](#)
- [Crea un utente con accesso amministrativo](#)
- [Crea una Account AWS chiave](#)

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 procedura consigliata in materia di sicurezza, assegna l'accesso amministrativo a un utente e utilizza solo l'utente root per eseguire [attività che richiedono l'accesso da parte dell'utente root](#).

AWS ti invia un'e-mail 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.

Crea un utente con accesso amministrativo

Dopo esserti registrato 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 [Signing in as the root user](#) 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.

Crea un utente con accesso amministrativo

1. Abilita Centro identità IAM.

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

2. In IAM Identity Center, concedi l'accesso amministrativo a un utente.

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

Accedi come utente con accesso amministrativo

- 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.

Assegna l'accesso ad altri utenti

1. In IAM Identity Center, crea un set di autorizzazioni che segua la migliore pratica di applicazione delle autorizzazioni con privilegi minimi.

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

2. Assegna gli utenti a un gruppo, quindi assegna l'accesso Single Sign-On al gruppo.

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

Crea una Account AWS chiave

È necessaria una Account AWS chiave per accedere ad Amazon Kinesis Video Streams a livello di codice.

Per creare una Account AWS chiave, procedi come segue:

1. Accedi AWS Management Console e apri la console IAM all'[indirizzo https://console.aws.amazon.com/iam/](https://console.aws.amazon.com/iam/).
2. Scegli Utenti nella barra di navigazione e scegli l'utente amministratore.
3. Seleziona la scheda Security credentials (Credenziali di sicurezza), quindi Create access key (Crea chiave di accesso).
4. Registra l'Access key ID (ID chiave di accesso) Scegli Show (Mostra) in Secret access key (Chiave di accesso segreta). Registra la Secret access key (Chiave di accesso segreta).

Creare un flusso video Kinesis

Questa sezione descrive come creare un flusso video Kinesis.

Questa sezione contiene le procedure seguenti:

- [the section called “Crea un flusso video utilizzando la console”](#)
- [the section called “Crea un flusso video utilizzando il AWS CLI”](#)

Crea un flusso video utilizzando la console

1. Apri la console all'[indirizzo https://console.aws.amazon.com/kinesisvideo/home](https://console.aws.amazon.com/kinesisvideo/home).
2. Nella pagina Video streams (Flussi video) scegliere Create video stream (Crea flusso video).
3. Nella pagina Crea un nuovo streaming video, inserisci *YourStreamName* il nome dello stream. Lascia selezionato il pulsante di configurazione predefinita.
4. Scegliere Create video stream (Crea flusso video).
5. Dopo che Amazon Kinesis Video Streams ha creato lo stream, consulta i dettagli *YourStreamName* sulla pagina.

Crea un flusso video utilizzando il AWS CLI

1. Verifica di avere AWS CLI installato e configurato il file. Per ulteriori informazioni, consulta la documentazione [AWS Command Line Interface](#).
2. Esegui il seguente comando Create-Stream nella AWS CLI:

```
aws kinesishvideo create-stream --stream-name "YourStreamName" --data-retention-in-hours 24
```

La risposta sarà simile alla seguente:

```
{
  "StreamARN": "arn:aws:kinesisvideo:us-
west-2:123456789012:stream/YourStreamName/123456789012"
}
```

Inviare dati a un flusso video di Amazon Kinesis

Questa sezione descrive come inviare dati multimediali da una videocamera al flusso video Kinesis creato nella sezione precedente. Questa sezione usa il [C++ Producer Library](#) come [Plugin GStreamer - kvssink](#) plugin.

Per inviare contenuti multimediali da una varietà di dispositivi su una varietà di sistemi operativi, questo tutorial utilizza la [libreria](#) di produttori Kinesis Video Streams C++ e GStreamer, un framework multimediale open source che standardizza l'accesso alle telecamere e ad altre fonti multimediali.

Argomenti

- [Crea l'SDK e gli esempi](#)
- [Esegui gli esempi per caricare contenuti multimediali su Kinesis Video Streams](#)
- [Rivedi gli oggetti di riconoscimento](#)

Crea l'SDK e gli esempi

Puoi creare l'SDK e gli esempi sul tuo computer o in AWS Cloud9 Segui le procedure appropriate riportate di seguito.

Build on your computer

Utilizzate le istruzioni contenute nel [file readme](#) per creare la libreria Producer e l'applicazione di esempio.

Questo include:

- Installazione delle dipendenze
- Clonazione del repository
- Usare CMake per generare makefile
- Creazione di file binari usando make

Build in AWS Cloud9

Segui queste procedure per caricare su Kinesis Video AWS Cloud9 Streams in. Non dovrai scaricare nulla sul tuo computer.

1. Nel AWS Management Console, aperto [AWS Cloud9](#).

Seleziona Crea ambiente.

2. Nella schermata Crea ambiente, completa quanto segue:
 - Nome: digita un nome per il nuovo ambiente.
 - Piattaforma: seleziona Ubuntu Server 22.04 LTS.

Puoi lasciare gli altri campi con le selezioni predefinite.

3. Una volta creato l'ambiente, seleziona Apri nella colonna IDE Cloud9.

Nella parte medio-bassa dello schermo, vedi. `Admin:~/environment $` Questo è il AWS Cloud9 terminale (Amazon EC2).

Note

Se chiudi accidentalmente il terminale, seleziona Finestra, Nuovo terminale.

Esegui i seguenti comandi nel terminale per modificare il volume a 20 GiB.

- a. Scarica lo script .

```
wget https://awsj-iot-handson.s3-ap-northeast-1.amazonaws.com/kvs-workshop/resize_volume.sh
```

- b. Assegna i permessi di esecuzione dello script.

```
chmod +x resize_volume.sh
```

- c. Eseguire lo script.

```
./resize_volume.sh
```

4. Recupera le informazioni più recenti su tutto il software che puoi installare o aggiornare tramite l'Advanced Packaging Tool (APT).

Questo comando non aggiorna il software stesso, ma assicura che il sistema sappia quali sono le ultime versioni disponibili.

```
sudo apt-get update
```

5. Installa le dipendenze C++ Producer SDK.

```
sudo apt-get install -y cmake m4 git build-essential pkg-config libssl-dev  
libcurl4-openssl-dev \  
liblog4cplus-dev libgstreamer1.0-dev libgstreamer-plugins-base1.0-dev \  
gstreamer1.0-plugins-base-apps gstreamer1.0-plugins-bad gstreamer1.0-plugins-  
good \  
gstreamer1.0-plugins-ugly gstreamer1.0-tools
```

6. Usa git per clonare l'SDK del produttore C++.

```
git clone https://github.com/aws-labs/amazon-kinesis-video-streams-producer-sdk-  
cpp.git
```

7. Prepara una directory di compilazione.

```
cd amazon-kinesis-video-streams-producer-sdk-cpp  
mkdir build  
cd build
```

8. Usa CMake per generare makefile.

```
cmake .. -DBUILD_GSTREAMER_PLUGIN=TRUE -DBUILD_DEPENDENCIES=OFF
```

La fine dell'output previsto è simile alla seguente:

```
-- Build files have been written to: /home/ubuntu/environment/amazon-kinesis-  
video-streams-producer-sdk-cpp/build
```

9. Usa `make` per compilare l'SDK e le applicazioni di esempio, oltre a creare gli eseguibili finali.

```
make
```

La fine dell'output previsto è simile alla seguente:

```
[100%] Linking CXX executable kvs_gstreamer_file_uploader_sample  
[100%] Built target kvs_gstreamer_file_uploader_sample
```

10. Conferma che i file di esempio sono stati creati. Elenca i file nella directory corrente:

```
ls
```

Verificate che siano presenti i seguenti file:

- `kvs_gstreamer_sample`
- `libgstkvssink.so`

11. (Facoltativo) È possibile aggiungere l'impostazione della variabile di ambiente `GST_PLUGIN_PATH` allo script di avvio della shell. Questo assicura che `GST_PLUGIN_PATH` sia impostato correttamente durante una nuova sessione di terminale. In AWS Cloud9, lo script di avvio della shell è: `~/.bashrc`

Esegui il comando seguente per aggiungere il comando alla fine dello script di avvio della shell.

```
echo "export GST_PLUGIN_PATH=~/environment/amazon-kinesis-video-streams-  
producer-sdk-cpp/build" >> ~/.bashrc
```

Digitate quanto segue per eseguire lo script di avvio della shell:

```
source ~/.bashrc
```

Conferma che `GST_PLUGIN_PATH` è impostato.

```
echo $GST_PLUGIN_PATH
```

Se impostate l'output correttamente, vedrete il seguente output. Se l'output è vuoto, la variabile di ambiente non è impostata correttamente.

```
/home/ubuntu/environment/amazon-kinesis-video-streams-producer-sdk-cpp/build
```

Esegui gli esempi per caricare contenuti multimediali su Kinesis Video Streams

L'applicazione di esempio non supporta le credenziali IMDS. Nel tuo terminale, esporta AWS le credenziali per il tuo utente o ruolo IAM e la regione in cui si trova lo stream.

```
export AWS_ACCESS_KEY_ID=YourAccessKey  
export AWS_SECRET_ACCESS_KEY=YourSecretKey  
export AWS_DEFAULT_REGION=YourAWSRegion
```

Se utilizzi AWS credenziali temporanee, esporta anche il token di sessione:

```
export AWS_SESSION_TOKEN=YourSessionToken
```

.mp4 files

Scarica un video di esempio in formato.mp4 da caricare su Kinesis Video Streams.

```
wget https://awsj-iot-handson.s3-ap-northeast-1.amazonaws.com/kvs-workshop/  
sample.mp4
```

Specifiche video:

- Risoluzione: 1280 x 720 pixel
- Frequenza fotogrammi: 30 fotogrammi al secondo
- Durata: 14,0 secondi
- Codifica video: H.264, nella traccia 1
- Fotogrammi chiave: ogni 3 secondi, con una durata del frammento (nota anche come dimensione di un gruppo di immagini (GoP)) di 3 secondi, con il frammento finale di 2 secondi.

Esegui il seguente comando con il nome dello stream che hai creato in precedenza. Se non hai ancora creato uno stream, consulta [the section called “Creare un flusso video Kinesis”](#).

```
./kvs_gstreamer_sample YourStreamName ./sample.mp4
```

Sample video from GStreamer

Usa il seguente comando per generare un video usando GStreamer.

Indica a GStreamer dove collocare il plugin GStreamer. `kvssink` Nella cartella di compilazione, specifica il percorso della cartella contenente il file. `libgstkvssink.so`

Dalla directory di compilazione, esegui il seguente comando:

```
export GST_PLUGIN_PATH=`pwd`
```

Questa pipeline GStreamer genera un flusso video di prova dal vivo con un pattern di test standard che viene eseguito a 10 fotogrammi al secondo con una risoluzione di 640x480 pixel. Viene aggiunto un overlay che mostra l'ora e la data correnti del sistema. Il video viene quindi codificato in formato H.264 e i fotogrammi chiave vengono generati al massimo ogni 10 fotogrammi, con una durata del frammento (nota anche come dimensione del gruppo di immagini (GoP)) di 1 secondo. `kvssink` prende il flusso video con codifica H.264, lo impacchetta nel formato contenitore Matroska (MKV) e lo carica nel flusso video Kinesis.

Esegui il comando seguente:

```
gst-launch-1.0 -v videotestsrc is-live=true \  
! video/x-raw,framerate=10/1,width=640,height=480 \  
! clockoverlay time-format="%a %B %d, %Y %I:%M:%S %p" \  
! x264enc bframes=0 key-int-max=10 \  
! h264parse \  
! kvssink stream-name="YourStreamName"
```

Per interrompere la pipeline GStreamer, seleziona la finestra del terminale e premi CTRL+C.

Note

Per ulteriori informazioni sull'utilizzo del plugin GStreamer per lo streaming di video da un flusso RTSP da una videocamera o da una videocamera USB, consulta. [Esempio: plug-in GStreamer SDK di Kinesis Video Streams Producer - kvssink](#)

Rivedi gli oggetti di riconoscimento

Durante il caricamento, Kinesis Video Streams invierà gli oggetti di riconoscimento al client che esegue il caricamento. Dovresti vederli stampati nell'output del comando. Un esempio è simile al seguente:

```
{"EventType": "PERSISTED", "FragmentTimecode": 1711124585823, "FragmentNumber": "1234567890123456789"}
```

Se la conferma EventType è PERSISTED, significa che Kinesis Video Streams ha archiviato e crittografato in modo duraturo questo blocco di file multimediali per il recupero, l'analisi e l'archiviazione a lungo termine.

Per ulteriori informazioni sui riconoscimenti, consulta. [the section called "PutMedia"](#)

Consuma dati multimediali

È possibile utilizzare i dati multimediali visualizzandoli nella console o creando un'applicazione che legge i dati multimediali da uno stream utilizzando Hypertext Live Streaming (HLS).

Visualizza i contenuti multimediali nella console

In un'altra scheda del browser, apri il AWS Management Console. Nella dashboard di Kinesis Video Streams, seleziona Streams video.

Seleziona il nome del tuo stream nell'elenco degli stream. Usa la barra di ricerca, se necessario.

Espandi la sezione Riproduzione multimediale. Se il video è ancora in fase di caricamento, verrà visualizzato. Se il caricamento è terminato, seleziona la doppia freccia sinistra.

Consuma i dati multimediali utilizzando HLS

È possibile creare un'applicazione client che utilizza i dati da un flusso video Kinesis utilizzando HLS. Per ulteriori informazioni su come creare un'applicazione che consuma i dati multimediali utilizzando HLS, consulta [the section called “Video Playback \(Riproduzione video\)”](#).

Agente Amazon Kinesis Video Streams Edge

Amazon Kinesis Video Streams offre un modo efficiente ed economico per connettersi alle telecamere IP nelle sedi dei clienti. Con Amazon Kinesis Video Streams Edge Agent, puoi registrare e archiviare localmente video dalle telecamere e trasmettere video al cloud secondo una pianificazione definita dal cliente per l'archiviazione, la riproduzione e l'elaborazione analitica a lungo termine.

Note

Per accedere ad Amazon Kinesis Video Streams Edge Agent, [compila](#) questo breve modulo.

Puoi scaricare Amazon Kinesis Video Streams Edge Agent e distribuirlo sui tuoi dispositivi di edge computing locali. Puoi anche distribuirli facilmente in contenitori Docker in esecuzione su istanze Amazon EC2. Dopo la distribuzione, puoi utilizzare l'API Amazon Kinesis Video Streams per aggiornare le configurazioni di registrazione video e caricamento su cloud. La funzionalità funziona con qualsiasi telecamera IP in grado di eseguire lo streaming tramite il protocollo RTSP. Non richiede alcuna implementazione di firmware aggiuntivo per le telecamere.

Offriamo le seguenti installazioni per Amazon Kinesis Video Streams Edge Agent:

- Come AWS IoT Greengrass V2 componente: puoi installare Amazon Kinesis Video Streams Edge Agent AWS IoT Greengrass come componente AWS IoT Greengrass su qualsiasi dispositivo certificato. Per ulteriori informazioni AWS IoT Greengrass, consulta la [AWS IoT Greengrass Version 2 Developer Guide](#).
- Sì AWS Snowball Edge: puoi eseguire Amazon Kinesis Video Streams Edge Agent su dispositivi Snowball Edge. Per ulteriori informazioni, consulta la [AWS Snowball Edge Developer Guide](#).
- In una AWS IoT distribuzione nativa: puoi installare Amazon Kinesis Video Streams Edge Agent in modo nativo su qualsiasi istanza di calcolo. Edge SDK lo utilizza [AWS IoT Core](#) per gestire l'edge tramite. [the section called "Flusso di video Amazon Kinesis"](#)

Per iniziare a usare Amazon Kinesis Video Streams Edge Agent, continua con le procedure appropriate riportate di seguito.

Argomenti

- [Operazioni dell'API Amazon Kinesis Video Streams Edge Agent](#)

- [Monitoraggio dell'agente Amazon Kinesis Video Streams Edge](#)
- [Esegui Amazon Kinesis Video Streams Edge Agent in modalità diversa AWS IoT Greengrass](#)
- [Implementa l'agente Amazon Kinesis Video Streams Edge su AWS IoT Greengrass](#)
- [Domande frequenti su Amazon Kinesis Video Streams Edge Agent](#)

Operazioni dell'API Amazon Kinesis Video Streams Edge Agent

Utilizza le seguenti operazioni API per configurare l'agente Amazon Kinesis Video Streams Edge:

- [the section called “StartEdgeConfigurationUpdate”](#)
- [the section called “DescribeEdgeConfiguration”](#)
- [the section called “DeleteEdgeConfiguration”](#)
- [the section called “ListEdgeAgentConfigurations”](#)

Monitoraggio dell'agente Amazon Kinesis Video Streams Edge

Per monitorare il tuo Amazon Kinesis Video Streams Edge Agent, consulta. [the section called “Monitoraggio dell'agente Amazon Kinesis Video Streams Edge con CloudWatch”](#)

Esegui Amazon Kinesis Video Streams Edge Agent in modalità diversa AWS IoT Greengrass

Segui questi passaggi per eseguire Amazon Kinesis Video Streams Edge AWS IoT Agent con MQTT come distribuzione autonoma.

Argomenti

- [Passaggio 1: installa le dipendenze necessarie sul dispositivo](#)
- [Fase 2: Crea gli Amazon Kinesis Video AWS Secrets Manager Streams e le risorse per gli URL RTSP della tua telecamera IP](#)
- [Fase 3: Creare una policy di autorizzazioni IAM](#)
- [Fase 4: Creare un ruolo IAM](#)
- [Fase 5: Creare l'alias del AWS IoT ruolo](#)
- [Fase 6: Creare la AWS IoT politica](#)

- [Passaggio 7: Crea qualsiasi AWS IoT cosa e ottieni le credenziali per AWS IoT Core](#)
- [Fase 8: creazione ed esecuzione di Amazon Kinesis Video Streams Edge Agent](#)
- [Passaggio 9: \(Facoltativo\) Installa l' CloudWatch agente sul dispositivo](#)
- [Fase 10: \(Facoltativo\) Esecuzione di Amazon Kinesis Video Streams Edge Agent come processo nativo](#)

Passaggio 1: installa le dipendenze necessarie sul dispositivo

Note

Per un elenco dei sistemi operativi supportati, consulta. [the section called “Quali sistemi operativi supporta Amazon Kinesis Video Streams Edge Agent?”](#)

Installa le dipendenze sul dispositivo

1. Per eseguire Amazon Kinesis Video Streams Edge Agent, installa le seguenti librerie appropriate sul tuo dispositivo:

Ubuntu

Type:

```
wget -O- https://apt.corretto.aws/corretto.key | sudo apt-key add -  
sudo add-apt-repository 'deb https://apt.corretto.aws stable main'  
sudo apt-get update  
  
sudo apt-get install -y gcc libssl-dev libcurl4-openssl-dev liblog4cplus-dev \  
libgstreamer1.0-dev libgstreamer-plugins-base1.0-dev \  
gstreamer1.0-plugins-base-apps gstreamer1.0-plugins-bad \  
gstreamer1.0-plugins-good gstreamer1.0-tools \  
unzip java-11-amazon-corretto-jdk maven
```

Amazon Linux 2

Type:

```
sudo yum update -y && sudo yum upgrade -y && sudo yum clean all -y  
sudo yum install -y gcc-c++ openssl-devel libcurl-devel gstreamer1* wget \  

```

```
java-11-amazon-corretto tar
```

Installa `log4cplus-2.1.0` dal codice sorgente.

```
wget https://github.com/log4cplus/log4cplus/releases/download/REL_2_1_0/
log4cplus-2.1.0.tar.gz
tar -xzvf log4cplus-2.1.0.tar.gz
cd log4cplus-2.1.0 && \
mkdir build && \
cd build && \
cmake .. && \
sudo make && \
sudo make install
```

Installa `apache-maven-3.9.2` dal codice sorgente.

```
wget https://dlcdn.apache.org/maven/maven-3/3.9.2/binaries/apache-maven-3.9.2-
bin.tar.gz
RUN tar -xzvf apache-maven-3.9.2-bin.tar.gz -C /opt
```

Important

Se vedi una schermata che ti dice che alcuni servizi devono essere riavviati, premi Invio per selezionare Ok.

Per ulteriori informazioni, consulta la [Guida per l'utente di Amazon Corretto 11](#).

2. Installa il AWS Command Line Interface. Consulta la sezione [Installazione o aggiornamento della versione più recente delle AWS CLI](#) procedure nella Guida AWS Command Line Interface per l'utente.

Fase 2: Crea gli Amazon Kinesis Video AWS Secrets Manager Streams e le risorse per gli URL RTSP della tua telecamera IP

Segui queste procedure per creare gli stream e i segreti necessari in AWS Secrets Manager. Eseguite innanzitutto questo passaggio, poiché nelle policy sono necessari gli ARN delle risorse create.

Crea flussi video Amazon Kinesis

Crea Amazon Kinesis Video Streams AWS Management Console utilizzando AWS CLI l'API, o.

Nel AWS Management Console, apri la console [Amazon Kinesis Video Streams](#). Scegli Streams video nella barra di navigazione a sinistra.

Per ulteriori informazioni, consulta [the section called "Creare un flusso video Kinesis"](#).

Crea segreti in AWS Secrets Manager

In AWS Management Console, apri la [AWS Secrets Manager console](#). Scegli Segreti nella barra di navigazione a sinistra.

Verifica che sia selezionata la regione appropriata.

1. Scegli Archivia un nuovo segreto.

a. Passaggio 1: scegli il tipo segreto

- Selezionare Other type of secret (Altro tipo di segreti).
- Nella sezione Coppie chiave/valore, aggiungi una coppia chiave-valore.

Chiave: MediaURI

Note

La chiave deve essere. MediaURI Fa distinzione tra maiuscole e minuscole. Se lo inserisci in modo errato, l'applicazione non funziona.

Valore:*Your MediaURI*.

Example

Esempio:rtsp://<YourCameraIPAddress>:<YourCameraRTSPPort>/
YourCameraMediaURI.

- a. Fase 2: Configurare il segreto. Dai un nome a questo segreto. Dategli il nome che volete.
- c. Fase 3: Configurazione della rotazione - opzionale. Seleziona Successivo.
- d. Fase 4: revisione. Scegli Store.

2. Se il tuo segreto non viene visualizzato immediatamente, seleziona il pulsante di aggiornamento.

Scegli il nome del tuo segreto. Prendi nota dell'ARN segreto.

3. Ripeti questa procedura per ogni MediaURI da cui desideri eseguire lo streaming.

Note

La AWS rete blocca alcune fonti RTSP pubbliche. Non puoi accedervi dall'interno dell'istanza Amazon EC2 o se non sei gestito mentre sei connesso alla VPN.

Important

L'URL RTSP della videocamera dovrebbe trasmettere video in formato h.264. La durata del frammento non deve superare il limite indicato in [the section called "Limiti di Producer SDK"](#)

Amazon Kinesis Video Streams Edge Agent supporta solo video.

Esegui `gst-discoverer-1.0 Your RtspUrl` per assicurarti che la videocamera sia raggiungibile dal dispositivo.

Salva gli ARN per tutti gli stream e i segreti che hai creato. Ti servono per il passaggio successivo.

Fase 3: Creare una policy di autorizzazioni IAM

Segui queste procedure per creare una policy IAM. Questa politica di autorizzazioni consente il controllo selettivo dell'accesso (un sottoinsieme di operazioni supportate) per una risorsa. AWS In questo caso, le AWS risorse sono i flussi video verso i quali desideri che Amazon Kinesis Video Streams Edge Agent esegua lo streaming. Le risorse includono anche i AWS Secrets Manager segreti che Amazon Kinesis Video Streams Edge Agent può recuperare. [Per ulteriori informazioni, consulta le politiche IAM.](#)

Crea una policy utilizzando l'editor di policy JSON

1. Accedi AWS Management Console e apri la console IAM all'[indirizzo https://console.aws.amazon.com/iam/](https://console.aws.amazon.com/iam/).
2. Nel pannello di navigazione a sinistra, seleziona Policy.

Se è la prima volta che selezioni Policy, verrà visualizzata la pagina Benvenuto nelle policy gestite. Seleziona Inizia.

3. Nella parte superiore della pagina, scegli Crea policy.
4. Nella sezione Editor di policy, scegli l'opzione JSON.
5. Inserisci il documento di policy JSON seguente:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:PutMetricData",
        "kinesisvideo:ListStreams",
        "iot:Connect",
        "iot:Publish",
        "iot:Subscribe",
        "iot:Receive"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "kinesisvideo:DescribeStream",
        "kinesisvideo:PutMedia",
        "kinesisvideo:TagStream",
        "kinesisvideo:GetDataEndpoint"
      ],
      "Resource": [
        "arn:aws:kinesisvideo:*:*:stream/streamName1/*",
        "arn:aws:kinesisvideo:*:*:stream/streamName2/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": "secretsmanager:GetSecretValue",
      "Resource": [
        "arn:aws:secretsmanager:*:*:secret:*",

```

```

        "arn:aws:secretsmanager:*:*:secret:*"
    ]
}
]
}

```

Note

Sostituisci `arn:aws:kinesisvideo:*:*:stream/streamName1/*` e `arn:aws:kinesisvideo:*:*:stream/streamName2/*` con gli ARN per i flussi video e sostituiscili `arn:aws:secretsmanager:*:*:secret:*` con gli ARN che contengono i segreti MediaURI in cui hai creato. [the section called “2. Crea risorse per gli URL RTSP della tua telecamera IP”](#) Usa gli ARN per i segreti a cui desideri che Amazon Kinesis Video Streams Edge Agent acceda.

6. Seleziona Successivo.

Note

È possibile alternare le opzioni dell'editor Visivo e JSON in qualsiasi momento. Se tuttavia si apportano modifiche o si seleziona Successivo nell'editor Visivo, IAM potrebbe ristrutturare la policy in modo da ottimizzarla per l'editor visivo. Per ulteriori informazioni, consulta la sezione [Ristrutturazione delle politiche nella Guida](#) per l'utente IAM.

7. Nella pagina Rivedi e crea, inserisci il nome della politica e una descrizione opzionale per la politica che stai creando. Rivedi Autorizzazioni definite in questa policy per visualizzare le autorizzazioni concesse dalla policy.

8. Seleziona Crea policy per salvare la nuova policy.

Fase 4: Creare un ruolo IAM

Il ruolo creato in questo passaggio può essere assunto da AWS IoT () per ottenere credenziali temporanee da AWS Security Token Service ()AWS STS. Questa operazione viene eseguita quando si eseguono richieste di autorizzazione delle credenziali da Amazon Kinesis Video Streams Edge Agent.

Crea il ruolo di servizio per Amazon Kinesis Video Streams (console IAM)

1. [Accedi AWS Management Console e apri la console IAM all'indirizzo https://console.aws.amazon.com/iam/](https://console.aws.amazon.com/iam/).
2. Nel pannello di navigazione della console IAM, scegliere Ruoli e quindi Crea ruolo.
3. Scegli il tipo di ruolo Custom Trust Policy e incolla la seguente policy:

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Principal": {
      "Service": "credentials.iot.amazonaws.com"
    },
    "Action": "sts:AssumeRole"
  }
}
```

4. Seleziona la casella accanto alla policy IAM in cui hai creato [the section called “3. Crea una politica di autorizzazioni IAM”](#).
5. Seleziona Successivo.
6. Inserisci il nome del ruolo o il suffisso del nome del ruolo per aiutarti a identificare lo scopo di questo ruolo.

Example

Esempio: KvsEdgeAgentRole

7. (Facoltativo) In Description (Descrizione), inserisci una descrizione per il nuovo ruolo.
8. (Facoltativo) Aggiungi metadati al ruolo allegando tag come coppie chiave/valore.

Per ulteriori informazioni sull'utilizzo dei tag in IAM, consulta [Tagging IAM resources nella IAM User Guide](#).

9. Rivedere il ruolo e scegliere Crea ruolo.

Fase 5: Creare l'alias del AWS IoT ruolo

Segui queste procedure per creare un alias di AWS IoT ruolo per il ruolo IAM in cui hai creato. [the section called “4. Creazione di un ruolo IAM”](#) Un alias di ruolo è un modello di dati alternativo che

rimanda al ruolo IAM. Una richiesta del fornitore di AWS IoT credenziali deve includere un alias di ruolo per indicare quale ruolo IAM assumere per ottenere credenziali temporanee da (). AWS Security Token Service AWS STS Per ulteriori informazioni, consulta [Come usare un certificato per ottenere un token di sicurezza](#).

Crea l'alias del AWS IoT ruolo

1. Accedi AWS Management Console e apri la AWS IoT Core console all'[indirizzo https://console.aws.amazon.com/iot/](https://console.aws.amazon.com/iot/).
2. Verifica che sia selezionata la regione appropriata.
3. Nella barra di navigazione a sinistra, seleziona Sicurezza, quindi scegli Alias di ruolo.
4. Scegli Crea alias di ruolo.
5. Inserisci un nome per l'alias del tuo ruolo.

Example

Esempio: KvsEdgeAgentRoleAlias

6. Nel menu a discesa Ruolo, seleziona il ruolo IAM in cui hai creato. [the section called “4. Creazione di un ruolo IAM”](#)
7. Scegli Crea. Nella pagina successiva, vedrai una nota che indica che l'alias del tuo ruolo è stato creato con successo.
8. Cerca e seleziona l'alias di ruolo appena creato. Prendi nota dell'alias Role ARN. Ne hai bisogno per la AWS IoT politica nella fase successiva.

Fase 6: Creare la AWS IoT politica

Segui queste procedure per creare una AWS IoT policy da allegare al certificato del dispositivo. Ciò fornisce le autorizzazioni necessarie alle AWS IoT funzionalità e consente l'assunzione dell'alias di ruolo utilizzando il certificato.

Con AWS IoT Core le policy, puoi controllare l'accesso al piano dati. AWS IoT Core Il piano AWS IoT Core dati è costituito da operazioni che è possibile utilizzare per eseguire le seguenti operazioni:

- Connect al broker di AWS IoT Core messaggi
- Inviare e ricevere messaggi MQTT
- Ottieni o aggiorna l'ombra del dispositivo di un oggetto

Per ulteriori informazioni, consulta [policy AWS IoT Core](#).

Usa l'editor di AWS IoT policy per creare una AWS IoT policy

1. Accedi AWS Management Console e apri la AWS IoT Core console all'[indirizzo https://console.aws.amazon.com/iot/](https://console.aws.amazon.com/iot/).
2. Nella barra di navigazione a sinistra, seleziona Sicurezza, quindi scegli Politiche.
3. Scegli Crea policy.
4. Immetti un nome da assegnare alla policy.

Example

Un esempio di nome di policy è KvsEdgeAccessIoTPolicy.

5. (Facoltativo) Aggiungere metadati alla policy collegando i tag come coppie chiave-valore.

Per ulteriori informazioni sull'utilizzo dei tag in IAM, consulta [Tagging your AWS IoT resources](#) nella AWS IoT Core Developer Guide.

6. Scegli la scheda JSON.
7. Incolla il seguente documento di policy JSON:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:Connect",
        "iot:Publish",
        "iot:Subscribe",
        "iot:Receive"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "iot:AssumeRoleWithCertificate"
      ],
```

```
        "Resource": "your-role-alias-arn"
      }
    ]
  }
```

Note

Sostituisci *your-role-alias-arn* con l'ARN dell'alias del ruolo in cui hai creato. [the section called “5. Creare l'alias del AWS IoT ruolo”](#)

8. Scegliete Crea per salvare il lavoro.

Passaggio 7: Crea qualsiasi AWS IoT cosa e ottieni le credenziali per AWS IoT Core

A questo punto hai creato:

- Una politica di autorizzazioni IAM. Per informazioni, consulta [the section called “3. Crea una politica di autorizzazioni IAM”](#).
- Un ruolo IAM, con la politica delle autorizzazioni allegata. Per informazioni, consulta [the section called “4. Creazione di un ruolo IAM”](#).
- Un alias di AWS IoT ruolo per il ruolo IAM. Per informazioni, consulta [the section called “5. Creare l'alias del AWS IoT ruolo”](#).
- Una AWS IoT politica, attualmente non associata a nessuna AWS risorsa. Per informazioni, consulta [the section called “6. Crea la AWS IoT politica”](#).

Per creare e registrare qualsiasi AWS IoT cosa e ottenere credenziali di AWS IoT Core accesso

1. Registra il dispositivo come AWS IoT oggetto e genera il certificato X.509 per il dispositivo.
 - a. [Accedere AWS Management Console e aprire la AWS IoT Core console all'indirizzo https://console.aws.amazon.com/iot/](https://console.aws.amazon.com/iot/).
 - b. Seleziona la regione appropriata.
 - c. Nella barra di navigazione a sinistra, seleziona Tutti i dispositivi, quindi scegli Cose.
 - d. Scegli Crea cose.
 - e. Seleziona Crea un singolo oggetto, quindi scegli Avanti.

1. Fase 1: Specificate le proprietà dell'oggetto

Digita un nome per l'oggetto, quindi scegli Avanti.

2. Fase 2. Configura il certificato del dispositivo

Seleziona Genera automaticamente un nuovo certificato (consigliato), quindi scegli Avanti.

3. Fase 3. Allega le politiche al certificato

Cerca la politica di autorizzazioni in [the section called "6. Crea la AWS IoT politica"](#) cui hai creato.

Seleziona la casella di controllo accanto alla tua politica e scegli Crea oggetto.

f. Nella finestra che appare, scarica i seguenti file:

- Certificato del dispositivo. Questo è il certificato X.509.
- File con chiave pubblica
- File con chiave privata
- Endpoint Amazon Trust Services (chiave RSA a 2048 bit: Amazon Root CA 1)

Prendi nota della posizione di ciascuno di questi file per un passaggio successivo.

g. Seleziona Fatto. Nella pagina successiva, vedrai una nota che indica che il file è stato creato con successo.

h. Trasferisci i file scaricati sopra sul tuo AWS IoT dispositivo, se non è già presente.

2. Ottieni l'endpoint del fornitore di credenziali per il tuo AWS account.

AWS CLI

Esegui il comando seguente:

```
aws iot describe-endpoint --endpoint-type iot:CredentialProvider
```

AWS Management Console

In [AWS CloudShell](#), esegui il seguente comando:

```
aws iot describe-endpoint --endpoint-type iot:CredentialProvider
```

Prendi nota di queste informazioni per un passaggio successivo.

3. Ottieni l'endpoint dei dati del dispositivo per il tuo AWS account.

AWS CLI

Esegui il comando seguente:

```
aws iot describe-endpoint --endpoint-type iot:Data-ATS
```

AWS Management Console

Esegui questa operazione:

1. Accedi AWS Management Console e apri la AWS IoT Core console all'[indirizzo https://console.aws.amazon.com/iot/](https://console.aws.amazon.com/iot/).
2. Nella barra di navigazione a sinistra, seleziona Impostazioni.
3. Individua l'endpoint dei dati del dispositivo.

Prendi nota di queste informazioni per un passaggio successivo.

4. (Facoltativo) Verifica che i certificati siano stati generati correttamente.

Esegui il comando seguente per verificare che i tuoi articoli siano stati generati correttamente.

```
curl --header "x-amzn-iot-thingname:your-thing-name" \  
  --cert /path/to/certificateID-certificate.pem.crt \  
  --key /path/to/certificateID-private.pem.key \  
  --cacert /path/to/AmazonRootCA1.pem \  
  https://your-credential-provider-endpoint/role-aliases/your-role-alias-name/  
  credentials
```

Per ulteriori informazioni, vedi [Come usare un certificato per ottenere un token di sicurezza](#).

Fase 8: creazione ed esecuzione di Amazon Kinesis Video Streams Edge Agent

Crea ed esegui Amazon Kinesis Video Streams Edge Agent

1. Scarica il tar file utilizzando il link che ti è stato fornito.

Se hai completato il modulo di interesse per Amazon Kinesis Video Streams Edge Agent, controlla la tua e-mail per il link per il download. [Se non hai completato il modulo, completalo qui.](#)

2. Verifica il checksum.
3. Estrai i file binari e il jar dal tuo dispositivo.

Tipo:tar -xvf kvs-edge-agent.tar.gz.

Dopo l'estrazione, la struttura delle cartelle sarà simile alla seguente:

```
kvs-edge-agent/LICENSE
kvs-edge-agent/THIRD-PARTY-LICENSES
kvs-edge-agent/pom.xml
kvs-edge-agent/KvsEdgeComponent
kvs-edge-agent/KvsEdgeComponent/recipes
kvs-edge-agent/KvsEdgeComponent/recipes/recipe.yaml
kvs-edge-agent/KvsEdgeComponent/artifacts
kvs-edge-agent/KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/edge_log_config
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/kvs-edge-agent.jar
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/libgstkvssink.so
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/libIngestorPipelineJNI.so
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/lib
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/lib/libcproducer.so
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/lib/libKinesisVideoProducer.so
```

Note

Il nome della cartella di rilascio deve essere impostato in modo da riflettere il numero della versione binaria più recente. Ad esempio, una versione 1.0.0 avrà il nome della cartella impostato come 1.0.0.

4. Crea il jar delle dipendenze.

Note

Il barattolo incluso con `kvs-edge-agent.tar.gz` non ha le dipendenze. Usa i seguenti passaggi per creare quelle librerie.

Vai alla `kvs-edge-agent` cartella che contiene `pom.xml`.

Tipo `mvn clean package`.

Questo genera un file jar contenente le dipendenze richieste da Amazon Kinesis Video Streams Edge Agent. `kvs-edge-agent/target/libs.jar`

5. Posiziona il file `libs.jar` nella cartella che contiene gli artefatti del componente.

Tipo `mv ./target/libs.jar ./KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/`.

6. Imposta le variabili di ambiente utilizzando i valori dei passaggi precedenti. La tabella seguente fornisce le descrizioni delle variabili.

Nome della variabile di ambiente	Richiesto	Descrizione
<code>AWS_REGION</code>	Sì	La regione utilizzata. Esempio: us-west-2
<code>AWS_IOT_CA_CERT</code>	Sì	Percorso del file del certificato CA utilizzato per stabilire

Nome della variabile di ambiente	Richiesto	Descrizione
		<p>la fiducia con il servizio di backend tramite TLS.</p> <p>Esempio: <i>/file/path/to/AmazonRootCA1.pem</i></p>
AWS_IOT_CORE_CERT	Sì	<p>Percorso del file del certificato X.509.</p> <p>Esempio: <i>/file/path/to/certificateID-certificate.pem.crt</i></p>
AWS_IOT_CORE_CREDENTIAL_ENDPOINT	Sì	<p>L'endpoint del provider di AWS IoT Core credenziali per il tuo account. AWS</p> <p>Esempio: <i>credential-account-specific-prefix.credentials.iot.amazonaws.com</i></p>
AWS_IOT_CORE_DATA_ENDPOINT	Sì	<p>L'endpoint del piano AWS IoT Core dati per il tuo account. AWS</p> <p>Esempio: <i>data-account-specific-prefix.iot.amazonaws.com</i></p>

Nome della variabile di ambiente	Richiesto	Descrizione
AWS_IOT_CORE_PRIVATE_KEY	Sì	<p>Percorso del file della chiave privata utilizzata nella coppia di chiavi pubblica/privata. Per ulteriori informazioni, vedere Gestione delle chiavi in. AWS IoT</p> <p>Esempio: <code>/file/path/to/certificateID-private.pem.key</code></p>
AWS_IOT_CORE_ROLE_ALIAS	Sì	<p>Il nome dell'alias del ruolo che punta al ruolo AWS IAM da utilizzare durante la connessione a. AWS IoT Core</p> <p>Esempio: <code>kvs-edge-role-alias</code></p>
AWS_IOT_CORE_THING_NAME	Sì	<p>Il nome dell' AWS IoT oggetto su cui viene eseguita l'applicazione.</p> <p>Esempio: <code>my-edge-device-thing</code></p>

Nome della variabile di ambiente	Richiesto	Descrizione
GST_PLUGIN_PATH	Sì	<p>Percorso del file che punta alla cartella che contiene le librerie gstreamer dipendenti dalla IngestorPipelineJNI piattaforma. Consente a GStreamer di caricare questi plugin. Per maggiori informazioni, consulta Scaricare, creare e configurare l'elemento GStreamer.</p> <p>Esempio: <i>/download-location /kvs-edge-agent/KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent/ EdgeAgent Version /</i></p>

Nome della variabile di ambiente	Richiesto	Descrizione
LD_LIBRARY_PATH	Sì	<p>Percorso del file che punta alla directory contenente le librerie <code>cproducer</code> e <code>KinesisVideoProducer</code> e quelle dipendenti dalla piattaforma.</p> <p>Esempio: <i>/download-location /kvs-edge-agent/KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent/ EdgeAgent Version /lib/</i></p>
AWS_KVS_EDGE_CLOUD_WATCH_ENABLED	No	<p>Determina se Amazon Kinesis Video Streams Edge Agent pubblicherà i parametri sullo stato del lavoro su Amazon CloudWatch</p> <p>Valori accettati: TRUE/FALSE (senza distinzione tra maiuscole e minuscole). Il valore predefinito è FALSE se non fornito.</p> <p>Esempio: FALSE</p>

Nome della variabile di ambiente	Richiesto	Descrizione
AWS_KVS_EDGE_LOG_LEVEL	No	<p>Il livello di registrazione degli output di Amazon Kinesis Video Streams Edge Agent.</p> <p>Valori accettati:</p> <ul style="list-style-type: none">• OFF• ALL• FATALI• ERRORE• WARN• INFO, impostazione predefinita, se non fornita• DEBUG• TRACE <p>Esempio: INFO</p>
AWS_KVS_EDGE_LOG_MAX_FILE_SIZE	No	<p>Una volta che il file di registro raggiunge questa dimensione e, si verificherà un rollover.</p> <ul style="list-style-type: none">• Minimo: 0• Massimo: 10000• Valore predefinito: 20, se non fornito• Unità: megabyte (MB) <p>Esempio: 5</p>

Nome della variabile di ambiente	Richiesto	Descrizione
AWS_KVS_EDGE_LOG_OUTPUT_DIRECTORY	No	<p>Il percorso del file che punta alla directory in cui vengono emessi i log di Amazon Kinesis Video Streams Edge Agent. Il valore predefinito è se non fornito. <code>./log</code></p> <p>Esempio: <code>/file/path/</code></p>
AWS_KVS_EDGE_LOG_ROLLOVER_COUNT	No	<p>Il numero di registri sovrapposti da conservare prima dell'eliminazione.</p> <ul style="list-style-type: none">• Minimo: 1• Massimo: 100• Valore predefinito: 10, se non fornito <p>Esempio: 20</p>
AWS_KVS_EDGE_RECORDING_DIRECTORY	No	<p>Verrà scritto il percorso del file che punta alla directory in cui vengono registrati i file multimediali. Il valore predefinito è la directory corrente se non viene fornita.</p> <p>Esempio: <code>/file/path/</code></p>

Nome della variabile di ambiente	Richiesto	Descrizione
GST_DEBUG	No	Specifica il livello dei log di GStreamer da emettere. Per ulteriori informazioni, consulta la documentazione di GStreamer. Esempio: 0
GST_DEBUG_FILE	No	Specifica il file di output dei log di debug di GStreamer. Se non è impostato, i log di debug vengono restituiti allo standard error. Per maggiori informazioni, consulta la documentazione di GStreamer. Esempio: <i>/tmp/gstreamer-logging.log</i>

7. Svuota la cache di GStreamer. Type:

```
rm ~/.cache/gstreamer-1.0/registry.your-os-architecture.bin
```

Per ulteriori informazioni, consulta la documentazione del registro di [GStreamer](#).

8. Preparare ed eseguire il comando java. L'agente Amazon Kinesis Video Streams Edge accetta i seguenti argomenti:

Nome della proprietà Java	Richiesto	Descrizione
java.library.path	No	Percorso del file che punta alla cartella contenente le librerie gstreamer e le librerie IngestorPipelineJNI dependent

Nome della proprietà Java

Richiesto

Descrizione

i. Se non vengono fornite, Amazon Kinesis Video Streams Edge Agent le cercherà nella directory corrente.

**Important**

L'agente Amazon Kinesis Video Streams Edge non funzionerà correttamente se non è in grado di individuare questi file.

Esempio: */file/path/*

Per impostarli, aggiungili `-Djava-property-name=value` al comando java usato per eseguire il jar.

Per esempio:

```
java -Djava.library.path=download-location/kvs-edge-agent/KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion \  
  --add-opens java.base/jdk.internal.misc=ALL-UNNAMED \  
  -Dio.netty.tryReflectionSetAccessible=true \  
  -cp kvs-edge-agent.jar:libs.jar \  
  com.amazonaws.kinesisvideo.edge.controller.ControllerApp
```

⚠ Important

Esegui il comando `java` precedente dalla stessa directory di `download-location/kvs-edge-agent/KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion`.

9. Invia le configurazioni all'applicazione utilizzando. AWS CLI

a. Crea un nuovo file, `example-edge-configuration.json`.

Incolla il codice seguente nel file. Questa è una configurazione di esempio che registra ogni giorno dalle 9:00 alle 16:59:59 (in base all'ora di sistema sul dispositivo). AWS IoT Inoltre, carica i file multimediali registrati ogni giorno dalle 19:00 alle 21:59:59.

Per ulteriori informazioni, consulta [the section called "StartEdgeConfigurationUpdate"](#).

```
{
  "StreamARN": "arn:aws:kinesisvideo:your-region:your-account-id:stream/your-stream/0123456789012",
  "EdgeConfig": {
    "HubDeviceArn": "arn:aws:iot:your-region:your-account-id:thing/kvs-edge-agent-demo",
    "RecorderConfig": {
      "MediaSourceConfig": {
        "MediaUriSecretArn": "arn:aws:secretsmanager:your-region:your-account-id:secret:your-secret-dRbHJQ",
        "MediaUriType": "RTSP_URI"
      },
      "ScheduleConfig": {
        "ScheduleExpression": "0 0 9,10,11,12,13,14,15,16 ? * * **",
        "DurationInSeconds": 3599
      }
    },
    "UploaderConfig": {
      "ScheduleConfig": {
        "ScheduleExpression": "0 0 19,20,21 ? * * **",
        "DurationInSeconds": 3599
      }
    },
    "DeletionConfig": {
      "EdgeRetentionInHours": 15,

```

```
    "LocalSizeConfig": {
      "MaxLocalMediaSizeInMB": 2800,
      "StrategyOnFullSize": "DELETE_OLDEST_MEDIA"
    },
    "DeleteAfterUpload": true
  }
}
```

- b. Per inviare il file ad Amazon Kinesis Video Streams Edge Agent, digita quanto segue nell'AWS CLI

```
aws kinesishvideo start-edge-configuration-update --cli-input-json
"file://example-edge-configuration.json"
```

10. Ripeti il passaggio precedente per ogni stream per Amazon Kinesis Video Streams Edge Agent.

Passaggio 9: (Facoltativo) Installa l' CloudWatch agente sul dispositivo

Note

Fai attenzione alle [CloudWatchquote](#).

Segui queste procedure per installare e configurare l' CloudWatch agente su cui caricare automaticamente i log generati da Amazon Kinesis Video Streams Edge Agent. CloudWatch

Per [le procedure](#) di installazione dell' CloudWatch agente sul tuo dispositivo, consulta la Amazon CloudWatch User Guide.

Quando viene richiesta la configurazione, seleziona una delle seguenti configurazioni.

Important

`file_path`Nelle configurazioni seguenti si presuppone che venga utilizzata la posizione di output di registrazione predefinita.

Il percorso del file utilizzato presuppone che tu stia eseguendo Amazon Kinesis Video Streams Edge Agent dalla posizione: `download-location/kvs-edge-agent/KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent/`*version*

- Per configurare l' CloudWatch agente per caricare i log e pubblicare i parametri di RAM e CPU del dispositivo, incolla quanto segue nel file di configurazione.

```
{
  "agent": {
    "run_as_user": "ubuntu",
    "metrics_collection_interval": 60
  },
  "metrics": {
    "metrics_collected": {
      "mem": {
        "measurement": [
          "mem_used_percent"
        ],
        "append_dimensions": {
          "IotThing": "YourIotThingName"
        }
      },
      "cpu": {
        "resources": [
          "*"
        ],
        "measurement": [
          "usage_active"
        ],
        "totalcpu": true,
        "append_dimensions": {
          "IotThing": "YourIotThingName"
        }
      }
    }
  },
  "logs": {
    "logs_collected": {
      "files": {
        "collect_list": [
          {
            "file_path": "download-location/kvs-edge-agent/KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent/version/log/java_kvs.log",
            "log_group_name": "/aws/kinesisvideo/EdgeRuntimeAgent",
            "log_stream_name": "YourIotThingName-java_kvs.log"
          },
          {
```



```

    {
      "file_path": "download-location/kvs-edge-agent/KvsEdgeComponent/
artifacts/aws.kinesisvideo.KvsEdgeComponent/version/log/cpp_kvs_streams.log*",
      "log_group_name": "/aws/kinesisvideo/EdgeRuntimeAgent",
      "log_stream_name": "YourIotThingName-cpp_kvs_streams.log"
    },
    {
      "file_path": "download-location/kvs-edge-agent/KvsEdgeComponent/
artifacts/aws.kinesisvideo.KvsEdgeComponent/version/log/cpp_kvssink.log*",
      "log_group_name": "/aws/kinesisvideo/EdgeRuntimeAgent",
      "log_stream_name": "YourIotThingName-cpp_kvssink.log"
    }
  ]
}
}
}
}
}

```

Fase 10: (Facoltativo) Esecuzione di Amazon Kinesis Video Streams Edge Agent come processo nativo

Configura Amazon Kinesis Video Streams Edge Agent come servizio systemd.

systemd è un gestore di sistemi e servizi su dispositivi Linux. systemd è il modo consigliato per gestire il processo, in quanto riavvierà Amazon Kinesis Video Streams Edge Agent nel caso in cui l'applicazione riscontri un errore o il dispositivo che esegue l'applicazione perda alimentazione.

Esegui questa operazione:

Esegui Amazon Kinesis Video Streams Edge Agent come processo nativo

1. Crea un nuovo file `/etc/systemd/system` e assegnagli un nome.

`aws.kinesisvideo.edge-runtime-agent.service`

Incolla quanto segue:

```

[Unit]
Description=AWS Kinesis Video Streams edge agent
After=network.target
StartLimitBurst=3
StartLimitInterval=30

```

```
[Service]
Type=simple
Restart=on-failure
RestartSec=10
WorkingDirectory=/download-location/kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion
Environment="GST_PLUGIN_PATH=/download-location/kvs-edge-agent/KvsEdgeComponent/
artifacts/aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion"
Environment="LD_LIBRARY_PATH=/download-location/kvs-edge-agent/KvsEdgeComponent/
artifacts/aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/lib"
...
Environment="AWS_IOT_CORE_DATA_ATS_ENDPOINT=data-account-specific-prefix.iot.aws-
region.amazonaws.com"
ExecStart=/usr/lib/jvm/java-11-amazon-corretto/bin/java --add-opens java.base/
jdk.internal.misc=ALL-UNNAMED -Dio.netty.tryReflectionSetAccessible=true -cp kvs-
edge-agent.jar:libs.jar com.amazonaws.kinesisvideo.edge.controller.ControllerApp

[Install]
WantedBy=multi-user.target
```

Per ulteriori informazioni sui parametri accettati dal file di configurazione del `systemd` servizio, consultate la [documentazione](#).

Note

Aggiungere le variabili di ambiente obbligatorie nella `...` posizione, come specificato [in the section called “8. Crea ed esegui Amazon Kinesis Video Streams Edge Agent”](#).

2. Ricarica i file di servizio per includere il nuovo servizio.

Tipo `sudo systemctl daemon-reload`.

3. Avviare il servizio.

Tipo `sudo systemctl start aws.kinesisvideo.edge-runtime-agent.service`.

4. Controlla lo stato del servizio Amazon Kinesis Video Streams Edge Agent per verificare che sia in esecuzione.

Tipo `sudo systemctl status aws.kinesisvideo.edge-runtime-agent.service`.

Di seguito è riportato un esempio dell'output che vedrai.

```
aws.kinesisvideo.edge-runtime-agent.service - AWS Kinesis Video Streams edge agent
  Loaded: loaded (/etc/systemd/system/aws.kinesisvideo.edge-runtime-agent.service; disabled; vendor preset: enabled)
  Active: active (running) since Thu 2023-06-08 19:15:02 UTC; 6s ago
  Main PID: 506483 (java)
  Tasks: 23 (limit: 9518)
  Memory: 77.5M
  CPU: 4.214s
  CGroup: /system.slice/aws.kinesisvideo.edge-runtime-agent.service
          ##506483 /usr/lib/jvm/java-11-amazon-corretto/bin/java -cp kvs-edge-agent.jar:libs.jar com.amazonaws.kinesisvideo.edge.controller.ControllerApp
```

5. Ispeziona i registri per eventuali errori.

Tipo `journalctl -e -u aws.kinesisvideo.edge-runtime-agent.service`.

6. Digita `systemctl --help` per visualizzare l'elenco completo delle opzioni con cui gestire il processo. `systemctl`

Di seguito sono riportati alcuni comandi comuni per gestire Amazon Kinesis Video Streams Edge Agent:

- Per riavviare, digita. `sudo systemctl restart aws.kinesisvideo.edge-runtime-agent.service`
- Per interrompere, digita `sudo systemctl stop aws.kinesisvideo.edge-runtime-agent.service`.
- Per avviarsi automaticamente a ogni riavvio del dispositivo, digita `sudo systemctl enable aws.kinesisvideo.edge-runtime-agent.service`.

Implementa l'agente Amazon Kinesis Video Streams Edge su AWS IoT Greengrass

Segui questi passaggi per distribuire Amazon Kinesis Video Streams Edge Agent AWS IoT Greengrass per registrare e caricare contenuti multimediali da telecamere IP.

Argomenti

- [Passaggio 1: creare un'istanza Ubuntu Amazon EC2](#)
- [Passaggio 2: configura il dispositivo AWS IoT Greengrass V2 principale sul dispositivo](#)

- [Fase 3: Crea Amazon Kinesis Video AWS Secrets Manager Streams e risorse per gli URL RTSP della tua telecamera IP](#)
- [Passaggio 4: Aggiungere le autorizzazioni al ruolo del servizio di scambio di token \(TES\)](#)
- [Passaggio 5: installare il componente AWS IoT Greengrass Secret Manager sul dispositivo](#)
- [Fase 6: distribuire il componente Amazon Kinesis Video Streams Edge AWS IoT Greengrass Agent sul dispositivo](#)
- [Passaggio 7: \(Facoltativo\) Installa il componente di gestione dei AWS IoT Greengrass registri sul dispositivo](#)

Passaggio 1: creare un'istanza Ubuntu Amazon EC2

Effettua le seguenti operazioni per creare un'istanza Ubuntu Amazon EC2.

Crea un'istanza Ubuntu Amazon EC2

1. [Accedi AWS Management Console e apri la console Amazon EC2 all'indirizzo https://console.aws.amazon.com/ec2/.](https://console.aws.amazon.com/ec2/)

Verifica che sia selezionata la regione appropriata.

2. Scegliere Launch Instance (Avvia istanza).

Completare i seguenti campi:

- Nome: digitate un nome per l'istanza.
 - Immagini dell'applicazione e del sistema operativo (Amazon Machine Image): seleziona Ubuntu.
 - Tipo di istanza: seleziona t2.large.
 - Login con coppia di chiavi: crea la tua coppia di chiavi.
 - Impostazioni di rete: mantieni le impostazioni predefinite.
 - Configurazione dello storage: aumenta il volume a 256 GiB.
 - Impostazioni avanzate: mantiene le impostazioni predefinite.
3. Avvia l'istanza e inserisci SSH.

Esegui questa operazione:

1. Seleziona Istanze nella barra di navigazione a sinistra, quindi seleziona l'ID dell'istanza.

2. Scegli Connect in alto a destra.
 3. Scegli il client SSH e segui le istruzioni sullo schermo.
 4. Apri un terminale e vai al .pem file scaricato (probabilmente in~/Downloads).
 5. La prima volta che segui queste procedure, riceverai il messaggio «L'autenticità dell'host (...) non può essere stabilita». Digita sì.
4. Installa le librerie di sistema per creare Amazon Kinesis Video Streams Edge Agent sull'istanza.

```
wget -O- https://apt.corretto.aws/corretto.key | sudo apt-key add -  
sudo add-apt-repository 'deb https://apt.corretto.aws stable main'  
  
sudo apt-get update  
  
sudo apt-get install -y gcc libssl-dev libcurl4-openssl-dev liblog4cplus-dev \  
libgstreamer1.0-dev libgstreamer-plugins-base1.0-dev \  
gstreamer1.0-plugins-base-apps gstreamer1.0-plugins-bad \  
gstreamer1.0-plugins-good gstreamer1.0-tools \  
unzip java-11-amazon-corretto-jdk maven
```

Important

Se vedi una schermata che ti dice che alcuni servizi devono essere riavviati, premi Invio per selezionare Ok.

Per ulteriori informazioni, consulta la [Guida per l'utente di Amazon Corretto 11](#).

Passaggio 2: configura il dispositivo AWS IoT Greengrass V2 principale sul dispositivo

Segui queste procedure per installare il software AWS IoT Greengrass core nucleus sull'istanza Amazon EC2.

Configura il dispositivo principale AWS IoT Greengrass

1. Accedi a AWS Management Console, <https://console.aws.amazon.com/iot/>.

Verifica che sia selezionata la regione appropriata.

2. Nella barra di navigazione a sinistra, seleziona Dispositivi Greengrass, Dispositivi Core.
3. Scegli Configura un dispositivo principale.
4. Completa i passaggi visualizzati sullo schermo.
 - Fase 1: Registrare un dispositivo Greengrass core. Digita un nome per il dispositivo.
 - Fase 2: Aggiungere a un gruppo di oggetti per applicare una distribuzione continua. Seleziona Nessun gruppo.
 - Fase 3: Installare il software Greengrass Core. Seleziona Linux.
 - Passaggio 3.1: Installare Java sul dispositivo

Java viene installato come parte di [the section called "1. Crea un'istanza di Ubuntu"](#). Tornate a questo passaggio se non avete ancora installato Java.

- Passaggio 3.2: Copia AWS le credenziali sul dispositivo

Apri l'bash/zsh opzione e incolla i comandi di esportazione nell'istanza Amazon EC2.

- Passaggio 3.3: Esegui il programma di installazione
 1. Copia ed esegui i comandi Scarica il programma di installazione ed Esegui il programma di installazione nell'istanza Ubuntu Amazon EC2.

 Note

Il comando Esegui il programma di installazione si aggiornerà automaticamente in base al nome scelto nel passaggio precedente.

2. Prendi nota del ruolo TES (Token Exchange Service) che è stato creato. perché sarà necessaria in seguito.

 Note

Per impostazione predefinita, il ruolo creato si chiama TokenExchangeRoleGreenGrassV2.

Fase 3: Crea Amazon Kinesis Video AWS Secrets Manager Streams e risorse per gli URL RTSP della tua telecamera IP

Segui queste procedure per creare gli stream e i segreti necessari in AWS Secrets Manager. Esegui innanzitutto questo passaggio, poiché nelle policy sono necessari gli ARN delle risorse create.

Crea flussi video Amazon Kinesis

Crea Amazon Kinesis Video Streams AWS Management Console utilizzando AWS CLI l'API, o.

Nel AWS Management Console, apri la console [Amazon Kinesis Video Streams](#). Scegli Streams video nella barra di navigazione a sinistra.

Per ulteriori informazioni, consulta [the section called "Creare un flusso video Kinesis"](#).

Crea segreti in AWS Secrets Manager

In AWS Management Console, apri la [AWS Secrets Manager console](#). Scegli Segreti nella barra di navigazione a sinistra.

Verifica che sia selezionata la regione appropriata.

1. Scegli Archivia un nuovo segreto.

a. Passaggio 1: scegli il tipo segreto

- Selezionare Other type of secret (Altro tipo di segreti).
- Nella sezione Coppie chiave/valore, aggiungi una coppia chiave-valore.

Chiave: MediaURI

Note

La chiave deve essere MediaURI. Fa distinzione tra maiuscole e minuscole. Se lo inserisci in modo errato, l'applicazione non funziona.

Valore: *Your MediaURI*.

Example

Esempio:`rtsp://<YourCameraIPAddress>:<YourCameraRTSPPort>/
YourCameraMediaURI.`

- b. Fase 2: Configurare il segreto. Dai un nome a questo segreto. Dategli il nome che volete.
 - c. Fase 3: Configurazione della rotazione - opzionale. Seleziona Successivo.
 - d. Fase 4: revisione. Scegli Store.
2. Se il tuo segreto non viene visualizzato immediatamente, seleziona il pulsante di aggiornamento.

Scegli il nome del tuo segreto. Prendi nota dell'ARN segreto.

3. Ripeti questa procedura per ogni MediaURI da cui desideri eseguire lo streaming.

Note

La AWS rete blocca alcune fonti RTSP pubbliche. Non puoi accedervi dall'interno dell'istanza Amazon EC2 o se non sei gestito mentre sei connesso alla VPN.

Important

L'URL RTSP della videocamera dovrebbe trasmettere video in formato h.264. La durata del frammento non deve superare il limite indicato in [the section called "Limiti di Producer SDK"](#)

Amazon Kinesis Video Streams Edge Agent supporta solo video.

Esegui `gst-discoverer-1.0 Your RtspUrl` per assicurarti che la videocamera sia raggiungibile dal dispositivo.

Salva gli ARN per tutti gli stream e i segreti che hai creato. Ti servono per il passaggio successivo.

Passaggio 4: Aggiungere le autorizzazioni al ruolo del servizio di scambio di token (TES)

Concedi il ruolo di servizio di scambio di token (TES) al dispositivo che assume le autorizzazioni per visualizzare i segreti. Ciò è necessario per il corretto funzionamento del AWS Secrets Manager AWS IoT Greengrass componente.

Aggiungi autorizzazioni al ruolo TES

1. Accedi AWS Management Console e apri la console IAM all'[indirizzo https://console.aws.amazon.com/iam/](https://console.aws.amazon.com/iam/).
2. Scegli Ruoli nella barra di navigazione a sinistra e cerca il ruolo TES che hai creato in precedenza nel processo.
3. Nel menu a discesa Aggiungi autorizzazioni, seleziona Allega politiche.
4. Scegli Crea policy.
5. Scorri verso il basso e seleziona Modifica.
6. Nell'editor delle policy, scegli JSON e modifica la policy.

Sostituisci la politica con la seguente:

Note

Sostituisci `arn:aws:kinesisvideo:*:*:stream/streamName1/*` e `arn:aws:kinesisvideo:*:*:stream/streamName2/*` con gli ARN per gli stream che hai creato in un passaggio precedente.

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

```

    "Effect": "Allow",
    "Action": [
        "kinesisvideo:DescribeStream",
        "kinesisvideo:PutMedia",
        "kinesisvideo:TagStream",
        "kinesisvideo:GetDataEndpoint"
    ],
    "Resource": [
        "arn:aws:kinesisvideo:*:*:stream/streamName1/*",
        "arn:aws:kinesisvideo:*:*:stream/streamName2/*"
    ]
  }
]
}

```

7. Nella pagina Add tags (Aggiungi tag), scegliere Next: Review (Successivo: rivedi).
8. Assegna un nome alla tua politica, quindi scegli Crea politica.

Un esempio di nome di politica è KvsEdgeAccessPolicy.

9. Chiudi la scheda e torna alla scheda in cui stavi allegando una politica al ruolo TES.

Scegli il pulsante di aggiornamento, quindi cerca la politica appena creata.

Seleziona la casella di controllo e scegli Allega politiche.

Nella schermata successiva, viene visualizzata una nota che indica che Policy è stata associata correttamente al ruolo.

10. Crea e allega un'altra policy, questa volta per i tuoi segreti.

Sostituisci la politica con la seguente:

Note

Sostituisci `arn:aws:secretsmanager:*:*:secret:*` con gli ARN contenenti i segreti MediaURI in cui hai creato. [the section called “3. Crea risorse per gli URL RTSP della tua telecamera IP”](#)

```

{
  "Version": "2012-10-17",
  "Statement": [

```

```

    {
      "Effect": "Allow",
      "Action": "secretsmanager:GetSecretValue",
      "Resource": [
        "arn:aws:secretsmanager:*:*:secret:*",
        "arn:aws:secretsmanager:*:*:secret:*"
      ]
    }
  ]
}

```

11. Crea e allega un'altra policy, questa volta per le metriche. Amazon CloudWatch Sostituisci la politica con la seguente:

```

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

```

Passaggio 5: installare il componente AWS IoT Greengrass Secret Manager sul dispositivo

Amazon Kinesis Video Streams Edge Agent AWS IoT Greengrass richiede prima l'installazione del componente Secret Manager sul dispositivo.

Installa il componente Secret Manager

1. Accedi AWS Management Console e apri la AWS IoT Core console all'[indirizzo https://console.aws.amazon.com/iot/](https://console.aws.amazon.com/iot/). Verifica che sia selezionata la regione appropriata.
2. Nella barra di navigazione a sinistra, scegli Greengrass devices, Deployments.

Scegli la distribuzione con lo stesso obiettivo dell'oggetto in cui abbiamo creato. [the section called “2. Configura il dispositivo AWS IoT Greengrass principale”](#)

3. Nel menu a discesa Azioni nell'angolo in alto a destra, scegli Rivedi.

Nel pop-up che appare, scegli Revise deployment.

4. Completa le seguenti sezioni:

- Fase 1: Specificare l'obiettivo. Seleziona Successivo.
- Fase 2: Seleziona i componenti.
 - Verificate che il componente AWS.GreenGrass.cli sia selezionato. Non disinstallate questo componente.
 - Attiva l'interruttore Mostra solo i componenti selezionati e cerca aws.greengrass.SecretManager.
 - Seleziona la casella accanto a aws.greengrass.SecretManager, quindi scegli Avanti.
- Fase 3: Configurare i componenti. Configura il componente AWS IoT Greengrass Secret Manager per scaricare i segreti dall' AWS IoT Greengrass ambiente.

Seleziona aws.greengrass.SecretManager componente, quindi scegli Configura componente.

Nella schermata che appare, aggiorna gli AWS Secrets Manager ARN nella casella Configurazione da unire.

Note

Sostituiscili `arn:aws:secretsmanager:*:*:secret:*` con gli ARN dei segreti in cui hai creato. [the section called “3. Crea risorse per gli URL RTSP della tua telecamera IP”](#)

```
{
  "cloudSecrets": [
    {
      "arn": "arn:aws:secretsmanager:*:*:secret:*"
    },
    {
      "arn": "arn:aws:secretsmanager:*:*:secret:*"
    }
  ]
}
```

```
]
}
```

Note

`cloudSecrets` è un elenco di oggetti con la chiave `arn`. Per ulteriori informazioni, [consulta la sezione Configurazione del gestore segreto](#) nella Guida per gli AWS IoT Greengrass Version 2 sviluppatori.

Al termine, seleziona Conferma, quindi scegli Avanti.

- Passaggio 4: configura le impostazioni avanzate. Seleziona Avanti.
 - Fase 5: Revisione. Seleziona Implementa.
5. Verifica che il AWS Secrets Manager componente e le autorizzazioni siano stati installati correttamente.

Nell'istanza Ubuntu Amazon EC2, digita `sudo /greengrass/v2/bin/greengrass-cli component details --name aws.greengrass.SecretManager` per verificare che il componente abbia ricevuto la configurazione aggiornata.

6. Ispeziona i log AWS IoT Greengrass principali.

Tipo `sudo less /greengrass/v2/logs/greengrass.log`.

Verifica la presenza di errori di distribuzione.

Se si è verificato un errore, modifica la distribuzione per rimuovere il `aws.greengrass.SecretManager` componente.

Digitare `sudo service greengrass restart` per riavviare il servizio AWS IoT Greengrass principale.

Se l'errore di distribuzione era correlato alla mancanza di autorizzazioni, consulta la [the section called "4. Aggiungere le autorizzazioni al ruolo TES"](#) sezione per assicurarti che il ruolo TES disponga delle autorizzazioni appropriate. Quindi, ripeti questa sezione.

Aggiorna i segreti sul componente AWS IoT Greengrass Secret Manager

Important

Il componente AWS IoT Greengrass Secret Manager recupera e memorizza nella cache i segreti solo quando la distribuzione viene aggiornata.

Per aggiornare i segreti sul componente AWS IoT Greengrass Secret Manager, segui i passaggi precedenti da 1 a 6, con la seguente modifica.

Fase 3: Configurare i componenti. Configura il componente AWS IoT Greengrass Secret Manager per scaricare i segreti dall' AWS IoT Greengrass ambiente.

Seleziona `aws.greengrass.SecretManagercomponente`, quindi scegli Configura componente.

Nella schermata che appare, incolla [""] la casella Reimposta percorsi e aggiorna gli AWS Secrets Manager ARN nella casella Configurazione da unire.

Per ulteriori informazioni, consulta [Reimpostare](#) gli aggiornamenti.

Fase 6: distribuire il componente Amazon Kinesis Video Streams Edge AWS IoT Greengrass Agent sul dispositivo

Implementa il componente Amazon Kinesis Video Streams Edge AWS IoT Greengrass Agent sul dispositivo

1. Scarica il `tar` file utilizzando il link fornito.

Se hai completato il modulo di interesse per Amazon Kinesis Video Streams Edge Agent, controlla la tua e-mail per il link per il download. [Se non hai completato il modulo, completalo qui.](#)

2. Verifica il checksum.
3. Estrai i file binari e il `jar` dal tuo dispositivo.

```
Tipo:tar -xvf kvs-edge-agent.tar.gz.
```

Dopo l'estrazione, la struttura delle cartelle sarà simile alla seguente:

```
kvs-edge-agent/LICENSE
kvs-edge-agent/THIRD-PARTY-LICENSES
kvs-edge-agent/pom.xml
kvs-edge-agent/KvsEdgeComponent
kvs-edge-agent/KvsEdgeComponent/recipes
kvs-edge-agent/KvsEdgeComponent/recipes/recipe.yaml
kvs-edge-agent/KvsEdgeComponent/artifacts
kvs-edge-agent/KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/edge_log_config

kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/kvs-edge-agent.jar
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/libgstkvssink.so
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/libIngestorPipelineJNI.so
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/lib
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/lib/libcproducer.so
kvs-edge-agent/KvsEdgeComponent/artifacts/
aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/lib/libKinesisVideoProducer.so
```

Note

Il nome della cartella di rilascio deve essere impostato in modo da riflettere il numero della versione binaria più recente. Ad esempio, una versione 1.0.0 avrà il nome della cartella impostato come 1.0.0.

4. Crea il jar delle dipendenze.

Note

Il jar incluso in `kvs-edge-agent.tar.gz` non ha le dipendenze. Usa i seguenti passaggi per creare quelle librerie.

Vai alla `kvs-edge-agent` cartella che contiene `pom.xml`.

Tipo `mvn clean package`.

Questo genererà un file jar contenente le dipendenze richieste da Amazon Kinesis Video Streams Edge Agent. `kvs-edge-agent/target/libs.jar`

- Inserisci il file `libs.jar` nella cartella che contiene gli artefatti del componente.

Tipo `mv ./target/libs.jar ./KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent/EdgeAgentVersion/`.

- Facoltativo. Configura le proprietà. Amazon Kinesis Video Streams Edge Agent accetta le seguenti variabili di ambiente AWS IoT Greengrass in modalità:

Nome della variabile di ambiente	Richiesto	Descrizione
<code>AWS_REGION</code>	Sì	La regione utilizzata. Esempio: <code>us-west-2</code> AWS IoT Greengrass Il software di base imposta automaticamente questo valore per te. Per ulteriori informazioni, consultate l'argomento di riferimento relativo alla variabile di ambiente Component nella Guida per gli AWS IoT Greengrass Version 2 sviluppatori.
<code>GST_PLUGIN_PATH</code>	Sì	Percorso del file che punta alla cartella contenente le librerie <code>gstkvssink</code> e quelle dipendenti dalla <code>IngestorPipelineJNI</code>

Nome della variabile di ambiente	Richiesto	Descrizione
LD_LIBRARY_PATH	Sì	<p>piattaforma. Questo permette a GStreamer di caricare questi plugin. Per maggiori informazioni, consulta Scaricare, creare e configurare l'elemento GStreamer.</p> <p>Esempio: <i>/download-location /kvs-edge-agent/KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent/ EdgeAgent Version /</i></p> <p>Percorso del file che punta alla directory contenente le librerie <code>cproducer</code> e <code>KinesisVideoProducer</code> e quelle dipendenti dalla piattaforma.</p> <p>Esempio: <i>/download-location /kvs-edge-agent/KvsEdgeComponent/artifacts/aws.kinesisvideo.KvsEdgeComponent/ EdgeAgent Version /lib/</i></p>

Nome della variabile di ambiente	Richiesto	Descrizione
AWS_KVS_EDGE_CLOUD_WATCH_ENABLED	No	<p>Determina se Amazon Kinesis Video Streams Edge Agent pubblicherà i parametri sullo stato del lavoro su Amazon CloudWatch</p> <p>Valori accettati :TRUE/FALSE(senza distinzione tra maiuscole e minuscole). Il valore predefinito è FALSE se non fornito.</p> <p>Esempio: FALSE</p>
AWS_KVS_EDGE_LOG_LEVEL	No	<p>Il livello di registrazione degli output di Amazon Kinesis Video Streams Edge Agent.</p> <p>Valori accettati:</p> <ul style="list-style-type: none">• OFF• ALL• FATALI• ERRORE• WARN• INFO, impostazione predefinita, se non fornita• DEBUG• TRACE <p>Esempio: INFO</p>

Nome della variabile di ambiente	Richiesto	Descrizione
AWS_KVS_EDGE_LOG_MAX_FILE_SIZE	No	<p>Una volta che il file di registro raggiunge questa dimensione, si verificherà un rollover.</p> <ul style="list-style-type: none"> • Minimo: 1 • Massimo: 100 • Valore predefinito: 20, se non fornito • Unità: megabyte (MB) <p>Esempio: 5</p>
AWS_KVS_EDGE_LOG_OUTPUT_DIRECTORY	No	<p>Il percorso del file che punta alla directory in cui vengono emessi i log di Amazon Kinesis Video Streams Edge Agent. Il valore predefinito è se non fornito. <code>./log</code></p> <p>Esempio: <code>/file/path/</code></p>
AWS_KVS_EDGE_LOG_ROLLOVER_COUNT	No	<p>Il numero di registri sovrapposti da conservare prima dell'eliminazione.</p> <ul style="list-style-type: none"> • Minimo: 1 • Massimo: 100 • Valore predefinito: 10, se non fornito <p>Esempio: 20</p>

Nome della variabile di ambiente	Richiesto	Descrizione
AWS_KVS_EDGE_RECORDING_DIRECTORY	No	<p>Verrà scritto il percorso del file che punta alla directory in cui vengono registrati i file multimediali. Il valore predefinito è la directory corrente se non viene fornita.</p> <p>Esempio: <i>/file/path/</i></p>
GREENGRASS_ROOT_DIRECTORY	No	<p>Il percorso del file alla AWS IoT Greengrass directory principale.</p> <p>Il valore predefinito è <code>/greengrass/v2/</code> se non viene fornito.</p> <p>Esempio: <i>/file/path/</i></p>
GST_DEBUG	No	<p>Specifica il livello dei log di GStreamer da emettere.</p> <p>Per ulteriori informazioni, consulta la documentazione di GStreamer.</p> <p>Esempio: 0</p>

Nome della variabile di ambiente	Richiesto	Descrizione
GST_DEBUG_FILE	No	<p>Specifica il file di output dei log di debug di GStreamer . Se non è impostato, i log di debug vengono restituiti allo standard error. Per maggiori informazioni, consulta la documentazione di GStreamer.</p> <p>Esempio: <code>/tmp/gstreamer-logging .log</code></p>

Apri `kvs-edge-agent/KvsEdgeComponent/recipes/recipe.yaml` e modifica lo script di esecuzione per aggiungere una qualsiasi delle variabili di ambiente precedenti.

Important

Assicuratevi che lo script di esecuzione modificato non contenga caratteri di tabulazione. Il software AWS IoT Greengrass principale non sarà in grado di leggere la ricetta.

7. Implementa il componente Amazon Kinesis Video Streams Edge Agent. AWS IoT Greengrass

Type:

```
sudo /greengrass/v2/bin/greengrass-cli deployment create \
  --recipeDir <download location>/kvs-edge-agent/KvsEdgeComponent/recipes/ \
  --artifactDir <download location>/kvs-edge-agent/KvsEdgeComponent/artifacts/ \
  --merge "aws.kinesisvideo.KvsEdgeComponent=EdgeAgentVersion"
```

Per ulteriori informazioni, consulta le seguenti sezioni della Guida per gli AWS IoT Greengrass Version 2 sviluppatori:

- [AWS IoT Greengrass Comandi CLI](#)
- [Distribuisci AWS IoT Greengrass i componenti sui dispositivi](#)

8. Invia le configurazioni all'applicazione utilizzando. AWS CLI

- a. Crea un nuovo file, *example-edge-configuration.json*.

Incolla il codice seguente nel file. Questa è una configurazione di esempio che registra ogni giorno dalle 9:00 alle 16:59:59 (in base all'ora di sistema sul dispositivo). AWS IoT Inoltre, carica i file multimediali registrati ogni giorno dalle 19:00 alle 21:59:59.

Per ulteriori informazioni, consulta [the section called "StartEdgeConfigurationUpdate"](#).

```
{
  "StreamARN": "arn:aws:kinesisvideo:your-region:your-account-id:stream/your-stream/0123456789012",
  "EdgeConfig": {
    "HubDeviceArn": "arn:aws:iot:your-region:your-account-id:thing/kvs-edge-agent-demo",
    "RecorderConfig": {
      "MediaSourceConfig": {
        "MediaUriSecretArn": "arn:aws:secretsmanager:your-region:your-account-id:secret:your-secret-dRbHJQ",
        "MediaUriType": "RTSP_URI"
      },
      "ScheduleConfig": {
        "ScheduleExpression": "0 0 9,10,11,12,13,14,15,16 ? * * **",
        "DurationInSeconds": 3599
      }
    },
    "UploaderConfig": {
      "ScheduleConfig": {
        "ScheduleExpression": "0 0 19,20,21 ? * * **",
        "DurationInSeconds": 3599
      }
    },
    "DeletionConfig": {
      "EdgeRetentionInHours": 15,
      "LocalSizeConfig": {
        "MaxLocalMediaSizeInMB": 2800,
        "StrategyOnFullSize": "DELETE_OLDEST_MEDIA"
      },
      "DeleteAfterUpload": true
    }
  }
}
```

```
}
```

- b. Digita quanto segue AWS CLI per inviare il file all'agente Amazon Kinesis Video Streams Edge:

```
aws kinesismedia start-edge-configuration-update --cli-input-json  
"file://example-edge-configuration.json"
```

9. Ripeti il passaggio precedente per ogni stream per Amazon Kinesis Video Streams Edge Agent.

Passaggio 7: (Facoltativo) Installa il componente di gestione dei AWS IoT Greengrass registri sul dispositivo

Note

[Sii consapevole delle quote. CloudWatch](#)

Segui queste procedure per configurare i log di Amazon Kinesis Video Streams Edge Agent su cui CloudWatch caricarli automaticamente AWS IoT Greengrass utilizzando il componente log manager.

Installa il componente log manager AWS IoT Greengrass

1. Verifica che il ruolo del AWS IoT Greengrass dispositivo disponga delle [autorizzazioni appropriate](#).
 - a. Accedi AWS Management Console e apri la console IAM all'[indirizzo https://console.aws.amazon.com/iam/](https://console.aws.amazon.com/iam/).
 - b. Fai clic su Ruoli nella barra di navigazione a sinistra.
 - c. Scegli il nome del ruolo TES creato in [the section called "2. Configura il dispositivo AWS IoT Greengrass principale"](#). Usa la barra di ricerca se necessario.
 - d. Seleziona la GreengrassV2TokenExchangeRoleAccess politica.
 - e. Seleziona la scheda JSON e verifica che la politica abbia il seguente aspetto:

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

```
        "Effect": "Allow",
        "Action": [
            "logs:CreateLogGroup",
            "logs:CreateLogStream",
            "logs:PutLogEvents",
            "logs:DescribeLogStreams",
            "s3:GetBucketLocation"
        ],
        "Resource": "*"
    }
]
```

- f. Se la `GreengrassV2TokenExchangeRoleAccess` policy non esiste o se mancano alcune autorizzazioni richieste, crea una nuova policy IAM con queste autorizzazioni e collegala al ruolo TES creato in [the section called “2. Configura il dispositivo AWS IoT Greengrass principale”](#)
2. [Accedi AWS Management Console e apri la AWS IoT Core console all'indirizzo https://console.aws.amazon.com/iot/](https://console.aws.amazon.com/iot/). Verifica che sia selezionata la regione appropriata.
3. Nella barra di navigazione a sinistra, scegli Greengrass devices, Deployments.

Scegli la distribuzione con lo stesso obiettivo dell'oggetto in cui hai creato. [the section called “2. Configura il dispositivo AWS IoT Greengrass principale”](#)
4. Nell'angolo in alto a destra, seleziona Azioni, quindi scegli Rivedi.

Nel pop-up che appare, scegli Revise deployment.
5. Completa le seguenti sezioni:
 - a. Fase 1: Specificare l'obiettivo. Seleziona Successivo.
 - b. Fase 2: Seleziona i componenti.
 - i. Verifica che il componente `aws.GreenGrass.cli` e `aws.greengrass` siano presenti. SecretManager componenti sono ancora selezionati.

 Important

Non disinstallate questi componenti.

- ii. Attiva l'interruttore Mostra solo i componenti selezionati e cerca `aws.greengrass`. LogManager.

- iii. Seleziona la casella accanto a `aws.greengrass.LogManager`, quindi scegli Avanti.
- c. Fase 3: Configurare i componenti. Configura il componente di gestione dei AWS IoT Greengrass log per caricare i log generati da Amazon Kinesis Video Streams Edge Agent.

Seleziona `aws.greengrass.LogManager` componente, quindi scegli Configura componente.

Nella schermata che appare, incolla la seguente configurazione del log manager nella casella Configurazione da unire.

```
{
  "logsUploaderConfiguration": {
    "componentLogsConfigurationMap": {
      "aws.kinesisvideo.KvsEdgeComponent/java_kvs.log": {
        "diskSpaceLimit": "100",
        "diskSpaceLimitUnit": "MB",
        "logFileDirectoryPath": "/greengrass/v2/work/
aws.kinesisvideo.KvsEdgeComponent/log",
        "logFileRegex": "java_kvs.log\\w*"
      },
      "aws.kinesisvideo.KvsEdgeComponent/cpp_kvs_edge.log": {
        "diskSpaceLimit": "100",
        "diskSpaceLimitUnit": "MB",
        "logFileDirectoryPath": "/greengrass/v2/work/
aws.kinesisvideo.KvsEdgeComponent/log",
        "logFileRegex": "cpp_kvs_edge.log\\w*"
      },
      "aws.kinesisvideo.KvsEdgeComponent/cpp_kvssink.log": {
        "diskSpaceLimit": "100",
        "diskSpaceLimitUnit": "MB",
        "logFileDirectoryPath": "/greengrass/v2/work/
aws.kinesisvideo.KvsEdgeComponent/log",
        "logFileRegex": "cpp_kvssink.log\\w*"
      },
      "aws.kinesisvideo.KvsEdgeComponent/cpp_kvs_streams.log": {
        "diskSpaceLimit": "100",
        "diskSpaceLimitUnit": "MB",
        "logFileDirectoryPath": "/greengrass/v2/work/
aws.kinesisvideo.KvsEdgeComponent/log",
        "logFileRegex": "cpp_kvs_streams.log\\w*"
      }
    }
  },
}
```

```
"periodicUploadIntervalSec": "1"  
}
```

Important

La `logFileDirectoryPath` configurazione precedente presuppone che venga utilizzata la posizione di output di registrazione predefinita.

Note

Per ulteriori informazioni su ciascuno dei parametri per la configurazione del gestore dei registri, consultate la sezione [Log manager](#) della Guida per gli AWS IoT Greengrass Version 2 sviluppatori.

Al termine, seleziona Conferma, quindi scegli Avanti.

- d. Passaggio 4: configura le impostazioni avanzate. Seleziona Avanti.
- e. Fase 5: Revisione. Seleziona Implementa.
6. Verificare che il componente e le autorizzazioni del gestore dei AWS registri siano stati installati correttamente.
7. Sull'istanza Ubuntu Amazon EC2, digita `sudo /greengrass/v2/bin/greengrass-cli component details --name aws.greengrass.LogManager` per verificare che il componente abbia ricevuto la configurazione aggiornata.
8. Ispeziona i log AWS IoT Greengrass principali.

Tipo `sudo less /greengrass/v2/logs/greengrass.log`.

Verifica la presenza di errori di distribuzione.

Se si è verificato un errore, modifica la distribuzione per rimuovere il `aws.greengrass.LogManager` componente.

Digitare `sudo service greengrass restart` per riavviare il servizio AWS IoT Greengrass principale.

Se l'errore di distribuzione era correlato alla mancanza di autorizzazioni, verifica [the section called "4. Aggiungere le autorizzazioni al ruolo TES"](#) che il ruolo TES disponga delle autorizzazioni appropriate. Quindi, ripeti questa sezione.

Domande frequenti su Amazon Kinesis Video Streams Edge Agent

Di seguito sono riportate alcune domande comuni per il servizio Amazon Kinesis Video Streams Edge Agent.

Quali sistemi operativi supporta Amazon Kinesis Video Streams Edge Agent?

Amazon Kinesis Video Streams Edge Agent attualmente supporta i seguenti sistemi operativi:

Ubuntu

- 2.x
 - AMD64
- 18.x
 - ARM

AL2

- amzn 2
 - AMD64 amazonlinux:2.0.20210219.0-amd64 (Snowball)

L'agente Amazon Kinesis Video Streams Edge supporta i file multimediali H.265?

Amazon Kinesis Video Streams Edge Agent supporta solo flussi elementari H.264.

L'agente Amazon Kinesis Video Streams Edge funziona in AL2?

Sì.

Come posso eseguire più stream all'interno dell'oggetto o del AWS IoT dispositivo?

Inviare un altro [the section called “StartEdgeConfigurationUpdate”](#) agli stessi HubDeviceArn ma diversi Amazon Kinesis Video AWS Secrets Manager Streams/ARN.

Come posso modificare un messaggio

StartEdgeConfigurationUpdate dopo che è stato inviato?

Inviare un aggiornamento [the section called “StartEdgeConfigurationUpdate”](#) allo stesso HubDeviceArn con lo stesso Amazon Kinesis Video Streams ARN. Quando l'applicazione riceve il messaggio da Amazon Kinesis Video Streams, sostituisce la configurazione precedente per quel flusso. Le modifiche verranno quindi apportate.

Hai qualche esempio in comune **ScheduleConfigs**?

L'agente Amazon Kinesis Video Streams Edge utilizza l'ora di sistema del dispositivo su cui è in esecuzione.

Descrizione	ScheduleExpression	DurationInSeconds
Registrazione 24 ore su 24, 7 giorni su 7, caricamento ogni ora	(null ScheduleConfig)	
Tutti i giorni dalle 9:00 alle 16:59:59	0 0 9-16 * * ? *	3599
9:00:00 - 16:59:59 nei giorni feriali	0 0 9-16 ? * 2-6 *	3599
	0 0 9-16 ? * 2,3,4,5,6 *	3599
	0 0 9-16 ? * MON-FRI *	3599
	0 0 9-16 ? * MON,TUE,WED,THU,FRI *	3599

Descrizione	ScheduleExpression	DurationInSeconds
Dalle 9:00 alle 16:59:59 nei fine settimana	0 0 9-16 ? * SAT,SUN *	3599
10:00:00 PM - 23:59:59 PM nei giorni feriali	0 0 22,23 ? * MON-FRI *	3599
Dalle 9:00 alle 10:00 tutti i giorni	0 0 9 * * ? *	3600
Tutti i giorni dalle 16:00 alle 17:59:59	0 0 16-17 * * ? *	3599

Per altri esempi, consulta la documentazione di [Quartz](#).

Esiste un limite massimo di streaming?

Amazon Kinesis Video Streams Edge Agent ha attualmente un limite rigido di 16 stream per dispositivo. Usa l'[the section called "DeleteEdgeConfiguration"](#) API per eliminare gli stream da un dispositivo. L'aggiornamento di una configurazione per lo stesso stream utilizzando il [the section called "StartEdgeConfigurationUpdate"](#) non aumenta il numero di stream del dispositivo.

Come posso riavviare un processo che ha avuto un errore?

Se si verifica un errore, Amazon Kinesis Video Streams Edge Agent tenterà di riavviare il processo. Tuttavia, con alcuni errori (come errori di configurazione), è necessario riavviare manualmente il processo.

Per determinare quali processi devono essere riavviati manualmente, consulta la FatalError metrica in [the section called "Monitoraggio dell'agente Amazon Kinesis Video Streams Edge con CloudWatch"](#)

Invia nuovamente il file [the section called "StartEdgeConfigurationUpdate"](#) per riavviare il job per lo stream.

Come posso monitorare lo stato del mio agente Amazon Kinesis Video Streams Edge?

Per ulteriori informazioni, consulta [the section called “Monitoraggio dell'agente Amazon Kinesis Video Streams Edge con CloudWatch”](#).

Streaming di video tramite un VPC

Questa versione beta è disponibile in anteprima nella regione Europa (Parigi), eu-west-3. Per accedere a questi componenti e alla nostra guida introduttiva, [inviaci un'e-mail](#).

Il servizio endpoint VPC di Amazon Kinesis Video Streams ti consente di trasmettere e utilizzare video attraverso la rete Amazon senza che i dati passino attraverso la rete Internet pubblica.

Per richiedere l'accesso, [inviaci un'e-mail](#) con le seguenti informazioni:

- ID account
- ARN in streaming
- ID VPC

Note

Potremmo impiegare fino a una settimana per aggiungerti al servizio.

Se non hai mai lavorato con gli endpoint VPC in passato, consulta le seguenti informazioni per acquisire familiarità con il concetto:

- [AWS PrivateLink sfondo](#)
- [Guida introduttiva a VPC](#)

Informazioni aggiuntive

Una volta aggiunto alla versione beta, ti invieremo via email un link con ulteriori informazioni su questa funzionalità.

Procedure per gli endpoint VPC

Quote

Le principali differenze tra le quote sono:

- Quota inferiore per tutte le API con larghezza di banda (2 mbps):
 - PutMedia
 - GetMedia
 - GetMediaForFragmentList
- 10 stream consentiti per cliente

Creare un endpoint

Una volta inseriti nell'elenco dei permessi, riceverai il nome del servizio endpoint VPC per Amazon Kinesis Video Streams. Assomiglierà a `com.amazonaws.region.kinesisvideo`

Crea un [endpoint VPC di interfaccia](#) per Amazon Kinesis Video Streams utilizzando la console Amazon VPC o (). AWS Command Line Interface AWS CLI

Nel, digita quanto segue AWS CLI:

```
aws ec2 create-vpc-endpoint \  
--vpc-id customer-provided-vpc-id \  
--service-name com.amazonaws.eu-west-2.kinesisvideo \  
--private-dns-enabled
```

Important

Il traffico all'interno del tuo VPC utilizzerà il DNS privato per il routing sull'endpoint. Se non lo abiliti, dovrai implementare la tua logica DNS. [Per ulteriori informazioni sul DNS privato, consulta AWS PrivateLink la documentazione.](#)

Per ulteriori informazioni sull' AWS CLI opzione, consulta [create-vpc-endpoint](#).

Controlla l'accesso agli endpoint

Puoi allegare una policy per gli endpoint al tuo endpoint VPC che controlla l'accesso ad Amazon Kinesis Video Streams. Questa policy specifica le informazioni riportate di seguito:

- il principale che può eseguire azioni,
- le azioni che possono essere eseguite e
- le risorse su cui è possibile eseguire le azioni.

Per ulteriori informazioni, consulta [Controllare l'accesso ai servizi con endpoint VPC utilizzando le policy degli endpoint](#) nella Guida. AWS PrivateLink

Di seguito è riportato un esempio di policy sugli endpoint per Amazon Kinesis Video Streams. Se collegata a un endpoint, questa policy nega l'accesso alle PutMedia azioni elencate a tutti i principali su tutte le risorse.

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Deny",
      "Action": [
        "kinesisvideo:PutMedia"
      ],
      "Resource": "*"
    }
  ]
}
```

Immagini in Kinesis Video Streams

Puoi usare le API e gli SDK di Amazon Kinesis Video Streams per estrarre immagini dai tuoi stream video. Puoi utilizzare queste immagini per applicazioni di riproduzione avanzate come miniature o pulizia avanzata, o per l'uso in pipeline di machine learning. Kinesis Video Streams offre l'estrazione delle immagini su richiesta tramite un'API o l'estrazione automatica delle immagini dai tag di metadati in un video importato.

Per informazioni sull'utilizzo del supporto gestito di Kinesis Video Streams per le immagini, vedere:

- [Generazione di immagini su richiesta \(GetImages\)](#)- Questa API consente ai clienti di estrarre una o più immagini dai video archiviati in Kinesis Video Streams.
- [Generazione automatica di immagini \(S3 Delivery\)](#) - Configura Kinesis Video Streams per estrarre automaticamente le immagini dai dati video in tempo reale in base ai tag di un video caricato e consegnare le immagini al bucket S3 specificato dal cliente.

Argomenti

- [Nozioni di base su GetImages](#)
- [Guida introduttiva alla consegna di Amazon S3](#)

Nozioni di base su GetImages

Il supporto gestito per le immagini offre un modo completamente gestito per ottenere immagini dai dati video trasmessi e archiviati in Kinesis Video Streams. È possibile utilizzare le immagini per eseguire carichi di lavoro di machine learning (ML), ad esempio il rilevamento di persone, animali domestici o veicoli. Le immagini possono anche essere utilizzate per aggiungere elementi interattivi alla riproduzione, come anteprime delle immagini per eventi in movimento e la pulizia di un video clip.

Per ulteriori informazioni su Get Images funzionalità, vedi [GetImages](#) nel Amazon Kinesis Video Streaming multimediale archiviato Guida di riferimento alle API.

Guida introduttiva alla consegna di Amazon S3

Attualmente, i clienti eseguono e gestiscono la propria pipeline di transcodifica delle immagini per creare immagini per vari scopi, come la pulizia, l'anteprima delle immagini, l'esecuzione di modelli ML sulle immagini e altro ancora. Kinesis Video Streams offre la capacità di transcodificare e fornire le

immagini. Kinesis Video Streams estrarrà automaticamente le immagini dai dati video in tempo reale sulla base di un tag e le consegnerà a un bucket S3 specificato dal cliente.

UpdateImageGenerationConfiguration

Per configurare uno streaming video Kinesis per abilitare la generazione di immagini su Amazon S3:

1. Crea unSecchio S3per la generazione di immagini in base ai tag aggiunti nell'SDK utilizzando la nuova API. Nota laTIPI S3, necessaria nella fase successiva, quando si aggiornano le configurazioni di generazione delle immagini per gli stream.
2. Crea un file JSON chiamatoupdate-image-generation-input.jsoncon il seguente contenuto come input.

```
{
  "StreamName": "TestStream",
  "ImageGenerationConfiguration":
  {
    "Status": "ENABLED",
    "DestinationConfig":
    {
      "DestinationRegion": "us-east-1",
      "Uri": "s3://bucket-name"
    },
    "SamplingInterval": 200,
    "ImageSelectorType": "PRODUCER_TIMESTAMP",
    "Format": "JPEG",
    "FormatConfig": {
      "JPEGQuality": "80"
    },
    "WidthPixels": 320,
    "HeightPixels": 240
  }
}
```

Puoi usare ilAWS CLIper invocare il[UpdateImageGenerationConfiguration](#)Funzionamento dell'API per aggiungere l'ARN Amazon S3 creato in precedenza e modificare lo stato inENABLED.

```
aws kinesishvideo update-image-generation-configuration \
--cli-input-json file://./update-image-generation-input.json \
```

Richiesta:

```
UpdateImageGenerationConfiguration HTTP/1.1
```

```
Method: 'POST'
Path: '/updateImageGenerationConfiguration'
Body: {
  StreamName: 'String', // Optional. Either stream name or arn should be passed
  StreamArn: 'String', // Optional. Either stream name or arn should be passed
  ImageGenerationConfiguration : {
    // required
    Status: 'Enum', // ENABLED | DISABLED,
    ImageSelectorType: 'Enum', // SERVER_TIMESTAMP | PRODUCER_TIMESTAMP..
    DestinationConfig: {
      DestinationRegion: 'String',
      Uri: string,
    },
    SamplingInterval: 'Number'//
    Format: 'Enum', // JPEG | PNG
    // Optional parameters
    FormatConfig: {
      'String': 'String',
    },
  },
  WidthPixels: 'Number', // 1 - 3840 (4k).
  HeightPixels: 'Number' // 1 - 2160 (4k).
}
```

Risposta:

```
HTTP/1.1 200
Content-type: application/json
Body: {
}
```

Note

È necessario almeno 1 minuto per avviare il flusso di lavoro di generazione delle immagini dopo l'aggiornamento della configurazione di generazione delle immagini. Attendere almeno 1 minuto prima di invocare `PutMedia` dopo la chiamata di aggiornamento.

DescribeImageGenerationConfiguration

Per visualizzare le configurazioni di generazione delle immagini già impostate per uno stream, i clienti possono creare un `DescribeImageGenerationConfiguration` richiesta, come segue.

Richiesta:

```
DescribeImageGenerationConfiguration HTTP/1.1
```

```
Method: 'POST'
Path: '/describeImageGenerationConfiguration'
Body: {
  StreamName: 'String', // Optional. Either stream name or arn should be passed
  StreamArn: 'String', // Optional. Either stream name or arn should be passed
}
```

Risposta:

```
HTTP/1.1 200
Content-type: application/json
Body: {
  ImageGenerationConfiguration : {
    Status: 'Enum',
    ImageSelectorType: 'Enum', // SERVER_TIMESTAMP | PRODUCER_TIMESTAMP
    DestinationConfig: {
      DestinationRegion: 'String'
      Uri: 'string',
    },
    SamplingInterval: 'Number',
    Format: 'Enum',
```

```
FormatConfig: {
    'String': 'String',
},
WidthPixels: 'Number',
HeightPixels: 'Number'
}
```

Per saperne di più su `DescribeImageGenerationConfiguration` funzionalità, vedi [DescribeImageGenerationConfiguration](#) nel Guida per gli sviluppatori di Amazon Kinesis Video Streams.

Etichette MKV del produttore

È possibile utilizzare Kinesis Video Streams Producer SDK per etichettare specifici frammenti di interesse esponendo un'operazione API nell'SDK. Per un esempio di tag, vedi [questo codice](#). Richiamando questa API, l'SDK aggiungerà un set di tag MKV predefiniti insieme ai dati del frammento. Kinesis Video Streams riconoscerà questi speciali tag MKV e avvierà il flusso di lavoro di generazione delle immagini in base alla configurazione di elaborazione delle immagini di quel flusso.

Tutti i metadati dei frammenti forniti insieme ai tag di generazione di immagini Amazon S3 verranno salvati come metadati Amazon S3.

Sintassi dei tag Producer MKV

```
|+ Tags
| + Tag
| // MANDATORY: Predefined MKV tag to trigger image generation for the fragment
| + Simple
| + Name: AWS_KINESISVIDEO_IMAGE_GENERATION

| // OPTIONAL: S3 prefix which will be set as prefix for generated image.
| + Simple
| + Name: AWS_KINESISVIDEO_IMAGE_PREFIX
| + String: image_prefix_in_s3 // 256 bytes max m

| // OPTIONAL: Key value pairs that will be persisted as S3 Image object metadata.
| + Simple
| + Name: CUSTOM_KEY_1 // Max 128 bytes
```

```

| + String: CUSTOM_VALUE_1 // Max 256 bytes
| + Simple
| + Name: CUSTOM_KEY_2 // Max 128 bytes
| + String: CUSTOM_VALUE_2 // Max 256 bytes

```

Aggiungere tag di metadati in Producer SDK utilizzando `PutEventMetaData`

La `PutEventMetaData` funzione aggiunge un file MKV associato a un evento. `PutEventMetaData` richiede due parametri. Il primo parametro è un evento il cui valore deriva da `STREAM_EVENT_TYPE` enum. Il secondo parametro, [pStreamEventMetadata](#), è facoltativo e può essere utilizzato per includere metadati aggiuntivi come coppie chiave-valore. È possibile aggiungere un limite di cinque coppie di metadati chiave-valore.

Limiti

La tabella seguente elenca le limitazioni associate ai tag dei metadati. Se il limite dei tag dei metadati è regolabile, puoi richiedere un aumento tramite il tuo account manager.

Limite	Valore massimo	Adattabile
Lunghezza del prefisso dell'immagine	256	no
Lunghezza della chiave dei metadati opzionale	128	no
Lunghezza del valore dei metadati opzionale	256	no
Numero massimo di metadati opzionali	10	sì

Metadati degli oggetti S3

Per impostazione predefinita, Kinesis Video Streams imposterà il numero di frammento, produttore, timestamp del server dell'immagine generata come metadati dell'oggetto Amazon S3. Se nei tag MKV vengono specificati dati di frammento aggiuntivi, tali tag verranno aggiunti anche ai metadati

degli oggetti di Amazon S3. L'esempio seguente mostra la sintassi corretta per i metadati degli oggetti Amazon S3.

```
{
  // KVS S3 object metadata
  x-amz-meta-aws_kinesisvideo_fragment_number : 'string',
  x-amz-meta-aws_kinesisvideo_producer_timestamp: 'number',
  x-amz-meta-aws_kinesisvideo_server_timestamp: 'number',

  // Optional key value pair sent as part of the MKV tags
  custom_key_1: custom_value_1,
  custom_key_2: custom_value_2,
}
```

Percorso dell'oggetto S3 (immagine)

L'elenco seguente mostra il formato corretto del percorso dell'oggetto e descrive ogni elemento all'interno del percorso.

Formato:

ImagePrefix_ID account_StreamName_ImageTimecode_ID casuale.estensione del file

1. ImagePrefix - Valore di `AWS_KINESISVIDEO_IMAGE_PREFIX`.
2. AccountID - ID account con cui viene creato lo stream.
3. StreamName - Nome dello stream per il quale viene generata l'immagine.
4. ImageTimecode - Codice temporale dell'epoca nel frammento in cui viene generata l'immagine.
5. RandomID - GUID casuale.
6. file-extension - JPG o PNG in base al formato dell'immagine richiesto.

Raccomandazioni sugli URI di Amazon S3 per proteggersi dalle limitazioni

Se scrivi migliaia di immagini su Amazon S3, c'è il rischio di una limitazione. Per ulteriori informazioni, vedere [Limiti delle richieste di invio del prefisso S3](#).

Un prefisso Amazon S3 inizia con un limite PUT di 3.500 richieste PUT al secondo e aumenterà gradualmente nel tempo per i prefissi unici. Evita di utilizzare date e ore come prefissi Amazon S3. I dati con codifica temporale influiranno su un prefisso alla volta e verranno inoltre modificati regolarmente, invalidando i precedenti incrementi di prefisso. Per consentire una scalabilità Amazon S3 più rapida e coerente, ti consigliamo di aggiungere un prefisso casuale, come un codice esadecimale o un UUID, all'URI di destinazione di Amazon S3. Ad esempio, i prefissi di codice esadecimale divideranno naturalmente le richieste in modo casuale tra 16 prefissi diversi (un prefisso per ogni carattere esadecimale univoco), il che consentirà di ricevere 56.000 richieste PUT al secondo dopo il ridimensionamento automatico di Amazon S3.

Notifiche in Kinesis Video Streams

Quando un frammento multimediale è disponibile per l'utilizzo, Kinesis Video Streams invia una notifica ai clienti utilizzando notifiche (Amazon Simple Notification Service Amazon SNS). L'argomento seguente spiega come iniziare a utilizzare le notifiche.

UpdateNotificationConfiguration

Utilizza questa operazione API per aggiornare le informazioni di notifica per uno stream. Per ulteriori informazioni sulla `UpdateNotificationConfiguration` funzionalità, consulta [UpdateNotificationConfiguration](#) la Amazon Kinesis Video Streams Developer Guide.

Note

È necessario almeno un minuto per avviare la notifica dopo l'aggiornamento della configurazione della notifica. Attendi almeno un minuto prima di richiamare `PutMedia` dopo la chiamata di aggiornamento.

DescribeNotificationConfiguration

Usa questa API per descrivere una configurazione di notifica allegata a uno stream. Per ulteriori informazioni sulla `DescribeNotificationConfiguration` funzionalità, consulta [DescribeNotificationConfiguration](#) la Amazon Kinesis Video Streams Developer Guide.

Tag MKV del produttore

Puoi utilizzare Kinesis Video Streams Producer SDK per etichettare frammenti specifici di interesse esponendo un'operazione API nell'SDK. [Guarda un esempio di come funziona in questa sezione di codice](#). Dopo aver chiamato questa API, l'SDK aggiungerà un set di tag MKV predefiniti insieme ai dati del frammento. Kinesis Video Streams riconoscerà questi tag MKV speciali e avvierà notifiche per i frammenti etichettati.

Tutti i metadati dei frammenti forniti insieme ai tag MKV di notifica verranno pubblicati come parte del payload tematico di Amazon SNS.

Sintassi per i tag MKV di Producer

```

|+ Tags
| + Tag
| // MANDATORY: Predefined MKV tag to trigger the notification for the fragment
| + Simple
| + Name: AWS_KINESISVIDEO_NOTIFICATION
| + String
| // OPTIONAL: Key value pairs that will be sent as part of the Notification payload
| + Simple
| + Name: CUSTOM_KEY_1 // Max 128 bytes
| + String: CUSTOM_VALUE_1 // Max 256 bytes
| + Simple
| + Name: CUSTOM_KEY_2 // Max 128 bytes
| + String: CUSTOM_VALUE_2 // Max 256 bytes

```

Limiti dei tag MKV

La tabella seguente elenca le limitazioni associate ai tag di metadati. Se il limite dei tag di metadati è regolabile, puoi richiedere un aumento tramite il tuo account manager.

Limite	Valore massimo	Regolabile
Lunghezza della chiave dei metadati opzionale	128	No
Lunghezza del valore dei metadati opzionale	256	No
Numero massimo di metadati opzionali	10	Sì

Payload tematico Amazon SNS

Qualsiasi notifica avviata tramite il flusso di lavoro precedente fornirà il payload dell'argomento Amazon SNS, come illustrato nell'esempio seguente. Questo esempio è un messaggio Amazon SNS

che si verifica dopo aver consumato i dati di notifica da una coda Amazon Simple Queue Service (Amazon SQS).

```
{
  "Type" : "Notification",
  "MessageId" : Message ID,
  "TopicArn" : SNS ARN,
  "Subject" : "Kinesis Video Streams Notification",
  "Message" : "{\"StreamArn\":\\Stream Arn,\\\"FragmentNumber\\\":\\Fragment Number,
  \\\"FragmentStartProducerTimestamp\\\":FragmentStartProducerTimestamp,
    \\\"FragmentStartServerTimestamp\\\":FragmentStartServerTimestamp,
  \\\"NotificationType\\\":\\\"PERSISTED\\\",\\\"NotificationPayload\\\":{\\ CUSTOM_KEY_1:
  \\CUSTOM_VALUE_1,
    \\CUSTOM_KEY_2:\\CUSTOM_VALUE_2}}\",
  "Timestamp" : "2022-04-25T18:36:29.194Z",
  "SignatureVersion" : Signature Version,
  "Signature" : Signature,
  "SigningCertURL" : Signing Cert URL,
  "UnsubscribeURL" : Unsubscribe URL
}
```

```
Subject: "Kinesis Video Streams Notification"
Message:
{
  "StreamArn":Stream Arn,
  "FragmentNumber":Fragment Number,
  "FragmentStartProducerTimestamp":Fragment Start Producer Timestamp,
  "FragmentStartServerTimestamp":Fragment Start Server Timestamp,
  "NotificationType":"PERSISTED",
  "NotificationPayload":{
    CUSTOM_KEY_1:CUSTOM_VALUE_1,
    CUSTOM_KEY_2:CUSTOM_VALUE_2
  }
}
```

Visualizzazione dei messaggi Amazon SNS

Non puoi leggere i messaggi direttamente da un argomento di Amazon SNS perché non esiste un'API per farlo. Per visualizzare i messaggi, iscriviti a una coda SQS all'argomento SNS o scegli qualsiasi altra destinazione supportata da [Amazon SNS](#). Tuttavia, l'opzione più efficiente per visualizzare i messaggi è usare Amazon SQS.

Per visualizzare i tuoi messaggi Amazon SNS utilizzando Amazon SQS

1. Crea una [coda Amazon SQS](#).
2. Da AWS Management Console, apri l'argomento Amazon SNS impostato come destinazione in `NotificationConfiguration`
3. Scegli Crea abbonamento, quindi scegli la coda Amazon SQS creata nel primo passaggio.
4. Esegui una `PutMedia` sessione con la configurazione di notifica abilitata e con i tag MKV di notifica aggiunti ai frammenti.
5. Scegli la coda Amazon SQS nella console Amazon SQS, quindi seleziona Invia e ricevi messaggi per la coda Amazon SQS.
6. Sondaggio per i messaggi. Questo comando dovrebbe mostrare tutte le notifiche generate dalla `PutMedia` sessione. Per informazioni sul polling, consulta [Amazon SQS short and long polling](#).

Sicurezza in Amazon Kinesis Video Streams

La sicurezza del cloud AWS è la massima priorità. In qualità di AWS cliente, trarrai vantaggio da un data center e da un'architettura di rete progettati per soddisfare i requisiti delle organizzazioni più sensibili alla sicurezza.

La sicurezza è una responsabilità condivisa tra AWS te e te. Il [modello di responsabilità condivisa](#) descrive questo come sicurezza del cloud e sicurezza nel cloud:

- Sicurezza del cloud: AWS è responsabile della protezione dell'infrastruttura che gestisce AWS i servizi nel AWS cloud. AWS ti fornisce anche servizi che puoi utilizzare in modo sicuro. L'efficacia della nostra sicurezza è regolarmente testata e verificata da revisori di terze parti come parte dei [programmi di conformità AWS](#). Per maggiori informazioni sui programmi di conformità che si applicano a Kinesis Video Streams [AWS](#), consulta la sezione [Servizi rientranti nell'ambito](#) del programma di conformità.
- Sicurezza nel cloud: la tua responsabilità è determinata dal AWS servizio che utilizzi. L'utente è anche responsabile per altri fattori, tra cui la riservatezza dei dati, i requisiti dell'azienda e leggi e normative applicabili.

Questa documentazione aiuta a capire come applicare il modello di responsabilità condivisa quando si utilizza Kinesis Video Streams. I seguenti argomenti mostrano come configurare Kinesis Video Streams per soddisfare i tuoi obiettivi di sicurezza e conformità. Imparerai anche come utilizzare altri AWS servizi che possono aiutarti a monitorare e proteggere le tue risorse Kinesis Video Streams.

Argomenti

- [Protezione dei dati in Kinesis Video Streams](#)
- [Controllo dell'accesso alle risorse Kinesis Video Streams tramite IAM](#)
- [Controllo dell'accesso alle risorse Kinesis Video Streams tramite AWS IoT](#)
- [Monitoraggio dei flussi video di Amazon Kinesis](#)
- [Convalida della conformità per Amazon Kinesis Video Streams](#)
- [Resilienza in Amazon Kinesis Video Streams](#)
- [Sicurezza dell'infrastruttura in Kinesis Video Streams](#)
- [Best practice di sicurezza per Kinesis Video Streams](#)

Protezione dei dati in Kinesis Video Streams

Puoi utilizzare la crittografia lato server (SSE) utilizzando le chiavi AWS Key Management Service (AWS KMS) per soddisfare severi requisiti di gestione dei dati crittografando i dati inattivi in Amazon Kinesis Video Streams.

Argomenti

- [Cos'è la crittografia lato server per Kinesis Video Streams?](#)
- [Considerazioni su costi, regioni e prestazioni](#)
- [Come posso iniziare a usare la crittografia lato server?](#)
- [Creazione e utilizzo di una chiave gestita dal cliente](#)
- [Autorizzazioni per l'utilizzo di una chiave gestita dal cliente](#)

Cos'è la crittografia lato server per Kinesis Video Streams?

La crittografia lato server è una funzionalità di Kinesis Video Streams che crittografa automaticamente i dati prima che vengano archiviati a riposo utilizzando una chiave specificata dall'utente. AWS KMS I dati vengono crittografati prima di essere scritti sul livello di archiviazione dello stream di Kinesis Video Streams e vengono decrittografati dopo essere stati recuperati dallo storage. Di conseguenza, i tuoi dati sono sempre crittografati quando sono inattivi all'interno del servizio Kinesis Video Streams.

Con la crittografia lato server, i produttori e i consumatori di streaming video Kinesis non devono gestire le chiavi KMS o le operazioni crittografiche. Se la conservazione dei dati è abilitata, i dati vengono automaticamente crittografati quando entrano ed escono da Kinesis Video Streams, quindi i dati inattivi vengono crittografati. AWS KMS fornisce tutte le chiavi utilizzate dalla funzionalità di crittografia lato server. AWS KMS semplifica l'uso di una chiave KMS per Kinesis Video Streams gestita AWS da, una chiave specificata dall' AWS KMS utente importata nel servizio. AWS KMS

Considerazioni su costi, regioni e prestazioni

Quando si applica la crittografia lato server, si è soggetti all'utilizzo delle AWS KMS API e ai costi delle chiavi. A differenza delle AWS KMS chiavi personalizzate, la chiave (Default) `aws/kinesis-video` KMS viene offerta gratuitamente. Tuttavia, devi comunque pagare i costi di utilizzo dell'API sostenuti da Kinesis Video Streams per tuo conto.

I costi di utilizzo delle API si applicano a ogni chiave KMS, incluse quelle personalizzate. I AWS KMS costi variano in base al numero di credenziali utente utilizzate dai produttori e dai consumatori di dati, poiché ogni credenziale utente richiede una chiamata API unica a AWS KMS

Di seguito vengono descritti i costi per risorsa:

Chiavi

- La chiave KMS per Kinesis Video Streams gestita AWS da (alias `aws/kinesis-video` =) è gratuita.
- Le chiavi KMS generate dall'utente sono soggette a costi. AWS KMS key Per ulteriori informazioni, consulta la sezione [Prezzi di AWS Key Management Service](#).

AWS KMS Utilizzo delle API

Le richieste API per generare nuove chiavi di crittografia dei dati o per recuperare chiavi di crittografia esistenti aumentano all'aumentare del traffico e sono soggette ai costi di AWS KMS utilizzo. Per ulteriori informazioni, consulta [AWS Key Management Service Prezzi: utilizzo](#).

Kinesis Video Streams genera richieste chiave anche quando la conservazione è impostata su 0 (nessuna conservazione).

Disponibilità della crittografia lato server per regione

La crittografia lato server dei flussi video Kinesis è disponibile in tutti i paesi in cui è disponibile Kinesis Regioni AWS Video Streams.

Come posso iniziare a usare la crittografia lato server?

La crittografia lato server è sempre abilitata su Kinesis Video Streams. Se al momento della creazione dello stream non viene specificata una chiave fornita dall'utente, viene utilizzata la Chiave gestita da AWS (fornita da Kinesis Video Streams).

Una chiave KMS fornita dall'utente deve essere assegnata a un flusso video Kinesis al momento della creazione. Non puoi assegnare una chiave diversa a uno stream utilizzando l'API in un secondo momento. [UpdateStream](#)

Puoi assegnare una chiave KMS fornita dall'utente a un flusso video Kinesis in due modi:

- Quando crei un flusso video Kinesis in AWS Management Console, specifica la chiave KMS nella scheda Crittografia della pagina Crea un nuovo flusso video.

- Quando crei uno stream video Kinesis utilizzando l'[CreateStream](#) API, specifica l'ID della chiave nel `KmsKeyId` parametro.

Creazione e utilizzo di una chiave gestita dal cliente

Questa sezione descrive come creare e utilizzare le proprie chiavi KMS anziché utilizzare la chiave amministrata da Amazon Kinesis Video Streams.

Creazione di una chiave gestita dal cliente

Per informazioni su come creare chiavi personalizzate, consulta [Creating Keys](#) nella AWS Key Management Service Developer Guide. Dopo aver creato le chiavi per l'account, il servizio Kinesis Video Streams restituisce queste chiavi nell'elenco delle chiavi gestite dal cliente.

Utilizzo di una chiave gestita dal cliente

Dopo aver applicato le autorizzazioni corrette ai consumatori, ai produttori e agli amministratori, puoi utilizzare chiavi KMS personalizzate per conto Account AWS tuo o altrui. Account AWS Tutte le chiavi KMS del tuo account vengono visualizzate nell'elenco delle chiavi gestite dal cliente sulla console.

Per utilizzare le chiavi KMS personalizzate che si trovano in un altro account, devi disporre delle autorizzazioni per utilizzare tali chiavi. Inoltre, occorre creare il flusso tramite l'API `CreateStream`. Non puoi utilizzare le chiavi KMS di account diversi negli stream creati nella console.

Note

Non si accede alla chiave KMS finché non viene `GetMedia` eseguita l'operazione `PutMedia` o. Tali operazioni producono i risultati seguenti:

- Se la chiave specificata non esiste, l'operazione `CreateStream` ha esito positivo, ma `PutMedia` `GetMedia` le operazioni sullo stream hanno esito negativo.
- Se utilizzi la chiave fornita (`aws/kinesis-video`), la chiave non è presente nel tuo account fino a quando non viene eseguita la prima `PutMedia` `GetMedia` operazione.

Autorizzazioni per l'utilizzo di una chiave gestita dal cliente

Prima di poter utilizzare la crittografia lato server con una chiave gestita dal cliente, è necessario configurare le politiche delle chiavi KMS per consentire la crittografia degli stream e la crittografia e la

decrittografia dei record di flusso. Per esempi e ulteriori informazioni sulle AWS KMS autorizzazioni, consulta Autorizzazioni [AWS KMS API: Actions and Resources Reference](#).

Note

L'uso della chiave di servizio predefinita per la crittografia non richiede l'applicazione di autorizzazioni IAM personalizzate.

Prima di utilizzare una chiave gestita dal cliente, verifica che i produttori e i consumatori di streaming video Kinesis (responsabili IAM) siano utenti nella policy di chiave AWS KMS predefinita. In caso contrario, le operazioni di lettura e scrittura dai flussi non riusciranno, causando perdite di dati, ritardi delle elaborazioni o blocchi delle applicazioni. Puoi gestire le autorizzazioni relative alle chiavi KMS tramite le policy IAM. Per ulteriori informazioni, consulta [Using IAM Policies](#) with AWS KMS

Esempio di autorizzazioni di produttore

I produttori di streaming video Kinesis devono avere l'`kms:GenerateDataKey` autorizzazione:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:GenerateDataKey"
      ],
      "Resource": "arn:aws:kms:us-west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    {
      "Effect": "Allow",
      "Action": [
        "kinesis-video:PutMedia",
      ],
      "Resource": "arn:aws:kinesis-video:*:123456789012:MyStream"
    }
  ]
}
```

Esempio di autorizzazioni per i consumatori

Gli utenti che utilizzano lo streaming video Kinesis devono avere l'`kms:Decrypt` autorizzazione:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:us-west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    {
      "Effect": "Allow",
      "Action": [
        "kinesis-video:GetMedia",
      ],
      "Resource": "arn:aws:kinesis-video:*:123456789012:MyStream"
    }
  ]
}
```

Controllo dell'accesso alle risorse Kinesis Video Streams tramite IAM

Puoi utilizzare AWS Identity and Access Management (IAM) con Amazon Kinesis Video Streams per controllare se gli utenti della tua organizzazione possono eseguire un'attività utilizzando operazioni specifiche dell'API Kinesis Video Streams e se possono utilizzare risorse specifiche. AWS

Per ulteriori informazioni su IAM, consulta:

- [AWS Identity and Access Management \(IAM\)](#)
- [Nozioni di base](#)
- [Guida per l'utente di IAM](#)

Indice

- [Sintassi delle policy](#)
- [Azioni per Kinesis Video Streams](#)
- [Amazon Resource Names \(ARN\) per Kinesis Video Streams](#)
- [Concedere ad altri account IAM l'accesso a un flusso video Kinesis](#)
- [Criteri di esempio per Kinesis Video Streams](#)

Sintassi delle policy

Una policy IAM è un documento JSON costituito da una o più dichiarazioni. Ogni dichiarazione è strutturata come segue:

```
{
  "Statement": [{
    "Effect": "effect",
    "Action": "action",
    "Resource": "arn",
    "Condition": {
      "condition": {
        "key": "value"
      }
    }
  ]
}
```

Una dichiarazione è costituita da diversi elementi:

- **Effetto:** l'effetto può essere Allow o Deny. Per impostazione predefinita, gli utenti non dispongono dell'autorizzazione per l'utilizzo di risorse e operazioni API, pertanto tutte le richieste vengono rifiutate. Un permesso esplicito sostituisce l'impostazione predefinita. Un rifiuto esplicito sovrascrive tutti i consensi.
- **Azione:** l'azione è l'azione API specifica per la quale si concede o si nega l'autorizzazione.
- **Risorsa:** la risorsa interessata dall'azione. Per specificare una risorsa nella dichiarazione, devi utilizzare il suo ARN (Amazon Resource Name).
- **Condizione:** le condizioni sono facoltative. Possono essere utilizzate per controllare quando è in vigore una policy.

Durante la creazione e la gestione delle policy IAM, ti consigliamo di utilizzare [IAM Policy Generator](#) e [IAM Policy Simulator](#).

Azioni per Kinesis Video Streams

In una dichiarazione di policy IAM, è possibile specificare qualsiasi operazione API per qualsiasi servizio che supporta IAM. Per Kinesis Video Streams, usa il seguente prefisso con il nome dell'azione API: `kinesisvideo:`. For example: `kinesisvideo:CreateStream`, `kinesisvideo:ListStreams` e `kinesisvideo:DescribeStream`.

Per specificare più azioni in una sola istruzione, separa ciascuna di esse con una virgola come mostrato di seguito:

```
"Action": ["kinesisvideo:action1", "kinesisvideo:action2"]
```

Puoi anche specificare più operazioni tramite caratteri jolly. Ad esempio, puoi specificare tutte le operazioni il cui nome inizia con la parola "Get" come segue:

```
"Action": "kinesisvideo:Get*"
```

Per specificare tutte le operazioni di Kinesis Video Streams, usa il carattere jolly asterisco (*) come segue:

```
"Action": "kinesisvideo:*"
```

Per l'elenco completo delle azioni dell'API Kinesis Video Streams, [consulta il riferimento all'API Kinesis Video Streams](#).

Amazon Resource Names (ARN) per Kinesis Video Streams

Ogni dichiarazione di policy IAM si applica alle risorse specificate utilizzando i relativi ARN.

Utilizza il seguente formato di risorse ARN per Kinesis Video Streams:

```
arn:aws:kinesisvideo:region:account-id:stream/stream-name/code
```

Per esempio:

```
"Resource": arn:aws:kinesisvideo:*:111122223333:stream/my-stream/0123456789012
```

Puoi ottenere l'ARN di uno stream usando. [DescribeStream](#)

Concedere ad altri account IAM l'accesso a un flusso video Kinesis

Potrebbe essere necessario concedere l'autorizzazione ad altri account IAM per eseguire operazioni sugli stream in Kinesis Video Streams. La seguente panoramica descrive i passi per concedere l'accesso ai flussi video su tutti gli account:

1. Ottieni l'ID account a 12 cifre dell'account a cui desideri concedere le autorizzazioni per eseguire operazioni sulla risorsa di streaming creata nel tuo account.

Esempio: nei seguenti passaggi, utilizzeremo 1111 come ID account per l'account a cui desideri concedere l'autorizzazione e 9999 come ID per Kinesis Video Streams

2. Crea una policy gestita da IAM nell'account proprietario dello stream (Δ 9999) che consenta il livello di accesso che desideri concedere.

Politica di esempio:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kinesisvideo:GetDataEndpoint",
        "kinesisvideo:DescribeStream",
        "kinesisvideo:PutMedia"
      ],
      "Resource": "arn:aws:kinesisvideo:us-west-2:999999999999:stream/custom-stream-name/1613732218179"
    }
  ]
}
```

Per altri esempi di policy per le risorse Kinesis Video Streams [Policy di esempio](#), consulta la sezione successiva.

3. Crea un ruolo nell'account proprietario dello stream (9999) e specifica l'account a cui desideri concedere le autorizzazioni (1111). Ciò aggiungerà un'entità attendibile al ruolo.

Esempio di politica affidabile:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::<111111111111>:root"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Allega la politica che hai creato nel passaggio precedente a questo ruolo.

Ora hai creato un ruolo nell'account 9999 che ha l'autorizzazione per operazioni come `DescribeStreamGetDataEndpoint`, e `PutMedia` su una risorsa di flusso ARN nella policy gestita. Questo nuovo ruolo si fida anche che l'altro account, 1111, assuma questo ruolo.

 Important

Prendi nota del ruolo ARN, ti servirà nella fase successiva.

4. Crea una politica gestita nell'altro account, 1111, che consenta l'`AssumeRole`azione sul ruolo che hai creato nell'account 9999 nel passaggio precedente. Dovrai menzionare il ruolo ARN del passaggio precedente.

Politica di esempio:

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": "sts:AssumeRole",
    "Resource": "arn:aws:iam::<999999999999>:role/CustomRoleName"
  }
}
```

- Allega la policy creata nel passaggio precedente a un'entità IAM, ad esempio un ruolo o un utente nell'account 1111. Questo utente ora ha il permesso di assumere un ruolo CustomRoleName nell'account 9999.

Le credenziali di questo utente chiamano l' AWS STS AssumeRoleAPI per ottenere le credenziali della sessione, che vengono successivamente utilizzate per chiamare le API Kinesis Video Streams sullo stream creato nell'account 9999.

```
aws sts assume-role --role-arn "arn:aws:iam::999999999999:role/CustomRoleName" --
role-session-name "kvs-cross-account-assume-role"
{
  "Credentials": {
    "AccessKeyId": "",
    "SecretAccessKey": "",
    "SessionToken": "",
    "Expiration": ""
  },
  "AssumedRoleUser": {
    "AssumedRoleId": "",
    "Arn": ""
  }
}
```

- Imposta la chiave di accesso, la chiave segreta e le credenziali di sessione in base al set precedente nell'ambiente.

```
set AWS_ACCESS_KEY_ID=
set AWS_SECRET_ACCESS_KEY=
set AWS_SESSION_TOKEN=
```

- Esegui le API Kinesis Video Streams per descrivere e ottenere l'endpoint di dati per lo stream nell'account 9999.

```
aws kinesismedia describe-stream --stream-arn "arn:aws:kinesisvideo:us-
west-2:999999999999:stream/custom-stream-name/1613732218179"
{
  "StreamInfo": {
    "StreamName": "custom-stream-name",
    "StreamARN": "arn:aws:kinesisvideo:us-west-2:999999999999:stream/custom-
stream-name/1613732218179",
    "KmsKeyId": "arn:aws:kms:us-west-2:999999999999:alias/aws/kinesisvideo",
    "Version": "abcd",
```

```

        "Status": "ACTIVE",
        "CreationTime": "2018-02-19T10:56:58.179000+00:00",
        "DataRetentionInHours": 24
    }
}

aws kinesismedia get-data-endpoint --stream-arn "arn:aws:kinesisvideo:us-
west-2:999999999999:stream/custom-stream-name/1613732218179" --api-name "PUT_MEDIA"
{
    "DataEndpoint": "https://s-b12345.kinesisvideo.us-west-2.amazonaws.com"
}

```

[Per step-by-step istruzioni generiche sulla concessione dell'accesso a più account, consulta Delegate Access Access Across Using IAM Roles. Account AWS](#)

Criteri di esempio per Kinesis Video Streams

I seguenti criteri di esempio mostrano come controllare l'accesso degli utenti a Kinesis Video Streams.

Example 1: Consenti agli utenti di ottenere dati da qualsiasi flusso video Kinesis

Questa policy consente a un utente o a un gruppo di eseguire le `ListTagsForStream` operazioni `DescribeStream` `GetDataEndpoint` `GetMediaListStreams`, e su qualsiasi flusso video Kinesis. Questa policy è appropriata per gli utenti che possono ottenere i dati da qualsiasi flusso di video.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kinesisvideo:Describe*",
        "kinesisvideo:Get*",
        "kinesisvideo:List*"
      ],
      "Resource": "*"
    }
  ]
}

```

Example 2: Consenti a un utente di creare un flusso video Kinesis e di scrivervi dati

Questa policy consente a un utente o a un gruppo di eseguire le operazioni `CreateStream` e `PutMedia`. Questa policy è appropriata per una fotocamera di sicurezza su cui è possibile creare un flusso di video e inviare dati.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kinesisvideo:CreateStream",
        "kinesisvideo:PutMedia"
      ],
      "Resource": "*"
    }
  ]
}
```

Example 3: Consenti a un utente l'accesso completo a tutte le risorse Kinesis Video Streams

Questa policy consente a un utente o a un gruppo di eseguire qualsiasi operazione di Kinesis Video Streams su qualsiasi risorsa. Questa policy è appropriata per gli amministratori.

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

Example 4: Consenti a un utente di scrivere dati su un flusso video Kinesis specifico

Questa policy consente a un utente o a un gruppo di scrivere dati su un determinato flusso di video. Questa policy è appropriata per un dispositivo in grado di inviare dati a un solo flusso.

```
{
  "Version": "2012-10-17",
```

```
"Statement": [  
  {  
    "Effect": "Allow",  
    "Action": "kinesisvideo:PutMedia",  
    "Resource": "arn:aws:kinesisvideo:us-west-2:123456789012:stream/  
your_stream/0123456789012"  
  }  
]
```

Controllo dell'accesso alle risorse Kinesis Video Streams tramite AWS IoT

Questa sezione descrive come consentire a un dispositivo (ad esempio una videocamera) di inviare dati audio e video solo a un particolare flusso video Kinesis. Puoi farlo utilizzando il provider di AWS IoT credenziali e un ruolo AWS Identity and Access Management (IAM).

I dispositivi possono utilizzare i certificati X.509 a cui connettersi AWS IoT utilizzando i protocolli di autenticazione reciproca TLS. Altri Servizi AWS (ad esempio, Kinesis Video Streams) non supportano l'autenticazione basata su certificati, ma possono essere AWS richiamati utilizzando AWS credenziali in formato Signature Version 4. L'algoritmo Signature Version 4 richiede in genere che il chiamante disponga di un ID di chiave di accesso e di una chiave di accesso segreta. AWS IoT dispone di un provider di credenziali che consente di utilizzare il certificato X.509 integrato come identità univoca del dispositivo per autenticare AWS le richieste (ad esempio, le richieste a Kinesis Video Streams). Ciò elimina la necessità di memorizzare un ID della chiave di accesso e una chiave di accesso segreta sul dispositivo.

Il provider di credenziali autentica un client (in questo caso, un Kinesis Video Streams SDK in esecuzione sulla videocamera a cui desideri inviare i dati a un flusso video) utilizzando un certificato X.509 ed emette un token di sicurezza temporaneo con privilegi limitati. Puoi utilizzare il token per firmare e autenticare qualsiasi AWS richiesta (in questo caso, una chiamata a Kinesis Video Streams). Per ulteriori informazioni, consulta [Autorizzazione delle chiamate dirette](#) ai servizi. AWS

Questo modo di autenticare le richieste della videocamera a Kinesis Video Streams richiede la creazione e la configurazione di un ruolo IAM e l'associazione di policy IAM appropriate al ruolo in modo che il fornitore di credenziali possa assumere AWS IoT il ruolo per tuo conto.

[Per ulteriori informazioni in merito, consulta la Documentazione. AWS IoT](#)[AWS IoT Core](#) Per ulteriori informazioni su IAM, consulta [AWS Identity and Access Management \(IAM\)](#).

Argomenti

- [AWS IoT ThingName come nome dello stream](#)
- [AWS IoT CertificateId come nome dello stream](#)
- [Usa AWS IoT le credenziali per eseguire lo streaming su un nome di stream codificato](#)

AWS IoT ThingName come nome dello stream

Argomenti

- [Fase 1: Creare qualsiasi tipo di AWS IoT oggetto e qualsiasi AWS IoT cosa](#)
- [Fase 2: Creare un ruolo IAM che venga assunto da AWS IoT](#)
- [Fase 3: Creare e configurare il certificato X.509](#)
- [Fase 4: Verifica le AWS IoT credenziali con lo streaming video Kinesis](#)
- [Fase 5: Implementazione di AWS IoT certificati e credenziali sul file system della videocamera e streaming dei dati nel flusso video](#)

Fase 1: Creare qualsiasi tipo di AWS IoT oggetto e qualsiasi AWS IoT cosa

In AWS IoT, una cosa è una rappresentazione di un dispositivo o di un'entità logica specifici. In questo caso, qualsiasi AWS IoT elemento rappresenta il flusso video Kinesis per cui desideri configurare il controllo degli accessi a livello di risorsa. Per creare un oggetto, devi innanzitutto creare un tipo di oggetto. AWS IoT È possibile utilizzare i tipi di AWS IoT oggetto per memorizzare informazioni di descrizione e configurazione comuni a tutti gli elementi associati allo stesso tipo di oggetto.

1. Il comando di esempio seguente crea un tipo di oggetto `kvs_example_camera`:

```
aws --profile default iot create-thing-type --thing-type-name kvs_example_camera >
iot-thing-type.json
```

2. Questo comando di esempio crea l'`kvs_example_camera_stream` oggetto del tipo di `kvs_example_camera` oggetto:

```
aws --profile default iot create-thing --thing-name kvs_example_camera_stream --
thing-type-name kvs_example_camera > iot-thing.json
```

Fase 2: Creare un ruolo IAM che venga assunto da AWS IoT

I ruoli IAM sono simili agli utenti, in quanto un ruolo è un' AWS identità con politiche di autorizzazione che determinano ciò che l'identità può e non può fare. AWS Un ruolo può essere assunto da chiunque ne abbia bisogno. Tuttavia, quando assumi un ruolo, vengono fornite le credenziali di sicurezza provvisorie per la sessione del ruolo.

Il ruolo creato in questa fase può essere assunto AWS IoT per ottenere credenziali temporanee dal servizio di token di sicurezza (STS) quando si eseguono richieste di autorizzazione delle credenziali dal client. In questo caso, il client è l'SDK Kinesis Video Streams in esecuzione sulla videocamera.

Eseguire la procedura seguente per creare e configurare questo ruolo IAM:

1. Crea un ruolo IAM.

Il comando di esempio seguente crea un ruolo IAM denominato `KVSCameraCertificateBasedIAMRole`:

```
aws --profile default iam create-role --role-name KVSCameraCertificateBasedIAMRole
--assume-role-policy-document 'file://iam-policy-document.json' > iam-role.json
```

Puoi utilizzare il seguente JSON della policy di attendibilità per `iam-policy-document.json`:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "credentials.iot.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

2. Successivamente, allega una politica di autorizzazioni al ruolo IAM che hai creato in precedenza. Questa politica di autorizzazioni consente il controllo selettivo dell'accesso (un sottoinsieme di operazioni supportate) per una risorsa. AWS In questo caso, la AWS risorsa è il flusso video a cui desideri che la videocamera invii i dati. In altre parole, una volta completate tutte le fasi di configurazione, questa telecamera sarà in grado di inviare dati solo a questo flusso video.

```
aws --profile default iam put-role-policy --role-name
KVSCameraCertificateBasedIAMRole --policy-name KVSCameraIAMPolicy --policy-
document 'file://iam-permission-document.json'
```

Puoi utilizzare la seguente policy IAM JSON per iam-permission-document.json:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kinesisvideo:DescribeStream",
        "kinesisvideo:PutMedia",
        "kinesisvideo:TagStream",
        "kinesisvideo:GetDataEndpoint"
      ],
      "Resource": "arn:aws:kinesisvideo:*:*:stream/${credentials-
iot:ThingName}/*"
    }
  ]
}
```

Tieni presente che questa politica autorizza le azioni specificate solo su un flusso video (AWS risorsa) specificato dal segnaposto (`${credentials-iot:}`). ThingName Questo segnaposto assume il valore dell'attributo AWS IoT thing ThingName quando il provider di AWS IoT credenziali invia il nome del flusso video nella richiesta.

- Successivamente, crea un Role Alias per il tuo ruolo IAM. L'alias del ruolo è un modello di dati alternativo che punta al ruolo IAM. Una richiesta del fornitore di AWS IoT credenziali deve includere un alias di ruolo per indicare quale ruolo IAM assumere per ottenere le credenziali temporanee dall'STS.

Il comando di esempio seguente crea un alias del ruolo denominato KvsCameraIoTRoleAlias.

```
aws --profile default iot create-role-alias --role-alias KvsCameraIoTRoleAlias --
role-arn $(jq --raw-output '.Role.Arn' iam-role.json) --credential-duration-seconds
3600 > iot-role-alias.json
```

4. Ora puoi creare la politica che consentirà di AWS IoT assumere un ruolo con il certificato (una volta allegato) utilizzando l'alias del ruolo.

Il seguente comando di esempio crea una politica per AWS IoT called `KvsCameraIoTPolicy`.

```
aws --profile default iot create-policy --policy-name KvsCameraIoTPolicy --policy-document 'file://iot-policy-document.json'
```

È possibile utilizzare il comando seguente per creare il `iot-policy-document` documento.json JSON:

```
cat > iot-policy-document.json <<EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:AssumeRoleWithCertificate"
      ],
      "Resource": "$(jq --raw-output '.roleAliasArn' iot-role-alias.json)"
    }
  ]
}
EOF
```

Fase 3: Creare e configurare il certificato X.509

La comunicazione tra un dispositivo (il tuo flusso video) AWS IoT è protetta tramite l'uso di certificati X.509.

1. Create il certificato a cui allegare la politica creata AWS IoT in precedenza.

```
aws --profile default iot create-keys-and-certificate --set-as-active --certificate-pem-outfile certificate.pem --public-key-outfile public.pem.key --private-key-outfile private.pem.key > certificate
```

2. Allega la politica per AWS IoT (`KvsCameraIoTPolicy` creata in precedenza) a questo certificato.

```
aws --profile default iot attach-policy --policy-name KvsCameraIoTPolicy --target
$(jq --raw-output '.certificateArn' certificate)
```

3. Allega AWS IoT thing (`kvs_example_camera_stream`) al certificato che hai appena creato:

```
aws --profile default iot attach-thing-principal --thing-name
kvs_example_camera_stream --principal $(jq --raw-output '.certificateArn'
certificate)
```

4. Per autorizzare le richieste tramite il provider di AWS IoT credenziali, è necessario l'endpoint delle AWS IoT credenziali, che è univoco per il proprio ID. Account AWS È possibile utilizzare il seguente comando per ottenere l'endpoint delle credenziali. AWS IoT

```
aws --profile default iot describe-endpoint --endpoint-type iot:CredentialProvider
--output text > iot-credential-provider.txt
```

5. Oltre al certificato X.509 creato in precedenza, è necessario disporre anche di un certificato CA per stabilire la fiducia con il servizio di backend tramite TLS. Puoi ottenere il certificato CA utilizzando il comando seguente:

```
curl --silent 'https://www.amazontrust.com/repository/SFSRootCAG2.pem' --output
cacert.pem
```

Fase 4: Verifica le AWS IoT credenziali con lo streaming video Kinesis

Ora puoi testare le AWS IoT credenziali che hai configurato finora.

1. In primo luogo, creare un Kinesis Video Stream con cui testare questa configurazione.

Important

Create uno stream video con un nome identico al nome dell' AWS IoT oggetto creato nel passaggio precedente (`kvs_example_camera_stream`).

```
aws kinesishvideo create-stream --data-retention-in-hours 24 --stream-name
kvs_example_camera_stream
```

2. Quindi, chiamate il provider delle AWS IoT credenziali per ottenere le credenziali temporanee:

```
curl --silent -H "x-amzn-iot-thingname:kvs_example_camera_stream" --cert
certificate.pem --key private.pem.key https://IOT_GET_CREDENTIAL_ENDPOINT/role-
aliases/KvsCameraIoTRoleAlias/credentials --cacert ./cacert.pem > token.json
```

Note

È possibile utilizzare il seguente comando per ottenere:
IOT_GET_CREDENTIAL_ENDPOINT

```
IOT_GET_CREDENTIAL_ENDPOINT=`cat iot-credential-provider.txt`
```

L'output JSON contiene AccessKey, SecretKey e SessionToken, che puoi utilizzare per accedere a Kinesis Video Streams.

3. Per il test, puoi utilizzare queste credenziali per richiamare l'API Kinesis Video DescribeStream Streams per il flusso video di esempio. `kvs_example_camera_stream`

```
AWS_ACCESS_KEY_ID=$(jq --raw-output '.credentials.accessKeyId' token.json)
AWS_SECRET_ACCESS_KEY=$(jq --raw-output '.credentials.secretAccessKey' token.json)
AWS_SESSION_TOKEN=$(jq --raw-output '.credentials.sessionToken' token.json) aws
kinesisvideo describe-stream --stream-name kvs_example_camera_stream
```

Fase 5: Implementazione di AWS IoT certificati e credenziali sul file system della videocamera e streaming dei dati nel flusso video

Note

I passaggi di questa sezione descrivono l'invio di file multimediali a un flusso video Kinesis da una videocamera che utilizza il [the section called "C++ Producer Library"](#)

1. Copia il certificato X.509, la chiave privata e il certificato CA generati nei passaggi precedenti nel file system della videocamera. Specificate i percorsi in cui sono archiviati questi file, il nome

dell'alias del ruolo e l'endpoint delle AWS IoT credenziali per l'esecuzione del `gst-launch-1.0` comando o dell'applicazione di esempio.

2. Il seguente comando di esempio utilizza l'autorizzazione AWS IoT del certificato per inviare video a Kinesis Video Streams:

```
gst-launch-1.0 rtspsrc location=rtsp://YourCameraRtspUrl short-header=TRUE !
  rtph264depay ! video/x-h264,format=avc,alignment=au ! h264parse ! kvssink stream-
  name="kvs_example_camera_stream" aws-region="YourAWSRegion" iot-certificate="iot-
  certificate,endpoint=credential-account-specific-prefix.credentials.iot.aws-
  region.amazonaws.com,cert-path=/path/to/certificate.pem,key-path=/path/to/
  private.pem.key,ca-path=/path/to/cacert.pem,role-aliases=KvsCameraIoTRoleAlias"
```

AWS IoT CertificateId come nome dello stream

Per rappresentare il tuo dispositivo (ad esempio, la tua videocamera) attraverso un AWS IoT oggetto, ma autorizzare un nome di stream diverso, puoi utilizzare l'AWS IoT `certificateId` attributo come nome dello stream e fornire le autorizzazioni Kinesis Video Streams sullo streaming che utilizza. AWS IoT I passaggi per eseguire questa operazione sono simili a quelli descritti in precedenza, con alcune modifiche.

- Modifica la politica delle autorizzazioni in base al tuo ruolo IAM (`iam-permission-document.json`) come segue:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kinesisvideo:DescribeStream",
        "kinesisvideo:PutMedia",
        "kinesisvideo:TagStream",
        "kinesisvideo:GetDataEndpoint"
      ],
      "Resource": "arn:aws:kinesisvideo:*:*:stream/${credentials-
        iot:AwsCertificateId}/*"
    }
  ]
}
```

Note

L'ARN risorsa utilizza l'ID certificato come segnaposto per il nome del flusso. L'autorizzazione IAM funzionerà quando utilizzi l'ID del certificato come nome dello stream. Ottieni l'ID del certificato dal certificato in modo da poterlo utilizzare come nome dello stream nella seguente descrizione della chiamata API stream.

```
export CERTIFICATE_ID=`cat certificate | jq --raw-output '.certificateId'`
```

- Verificare questa modifica utilizzando il comando CLI describe-stream di Kinesis Video Streams:

```
AWS_ACCESS_KEY_ID=$(jq --raw-output '.credentials.accessKeyId' token.json)
AWS_SECRET_ACCESS_KEY=$(jq --raw-output '.credentials.secretAccessKey' token.json)
AWS_SESSION_TOKEN=$(jq --raw-output '.credentials.sessionToken' token.json) aws
kinesisvideo describe-stream --stream-name ${CERTIFICATE_ID}
```

- Passa il CertificateID al provider di credenziali AWS IoT nell'applicazione di [esempio nell'SDK Kinesis](#) Video Streams C++:

```
credential_provider =
    make_unique<IotCertCredentialProvider>(iot_get_credential_endpoint,
        cert_path,
        private_key_path,
        role_alias,
        ca_cert_path,
        certificateId);
```

Note

Nota che stai passando il thingname al provider delle credenziali. AWS IoT È possibile utilizzare getenv per passare il thingname all'applicazione demo in modo simile al passaggio degli altri attributi. AWS IoT Utilizzare l'ID certificato come nome del flusso nei parametri della riga di comando quando si esegue l'applicazione di esempio.

Usa AWS IoT le credenziali per eseguire lo streaming su un nome di stream codificato

Per rappresentare il tuo dispositivo (ad esempio, la tua videocamera) attraverso un AWS IoT oggetto, ma autorizzare lo streaming verso uno specifico flusso video di Amazon Kinesis, fornisci le autorizzazioni di Amazon Kinesis Video Streams sullo streaming che utilizza. AWS IoT Il processo è simile alle sezioni precedenti, con alcune modifiche.

Modifica la politica delle autorizzazioni in base al tuo ruolo IAM (`iam-permission-document.json`) come segue:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kinesisvideo:DescribeStream",
        "kinesisvideo:PutMedia",
        "kinesisvideo:TagStream",
        "kinesisvideo:GetDataEndpoint"
      ],
      "Resource": "arn:aws:kinesisvideo:*:*:stream/YourStreamName/*"
    }
  ]
}
```

Copia il certificato X.509, la chiave privata e il certificato CA generati nei passaggi precedenti nel file system della fotocamera.

Specificate i percorsi in cui sono archiviati questi file, il nome dell'alias del ruolo, il nome dell' AWS IoT oggetto e l'endpoint delle AWS IoT credenziali per l'esecuzione del `gst-launch-1.0` comando o dell'applicazione di esempio.

Il seguente comando di esempio utilizza l'autorizzazione del AWS IoT certificato per inviare video ad Amazon Kinesis Video Streams:

```
gst-launch-1.0 rtspsrc location=rtsp://YourCameraRtspUrl short-header=TRUE !
  rtph264depay ! video/x-h264,format=avc,alignment=au ! h264parse ! kvssink
  stream-name="YourStreamName" aws-region="YourAWSRegion" iot-certificate="iot-
```

```
certificate,endpoint=credential-account-specific-prefix.credentials.iot.aws-region.amazonaws.com,cert-path=/path/to/certificate.pem,key-path=/path/to/private.pem.key,ca-path=/path/to/cacert.pem,role-aliases=KvsCameraIoTRoleAlias,iot-thing-name=YourThingName"
```

Monitoraggio dei flussi video di Amazon Kinesis

Kinesis Video Streams offre funzionalità di monitoraggio per i flussi di distribuzione. Per ulteriori informazioni, consulta [Monitoraggio](#).

Convalida della conformità per Amazon Kinesis Video Streams

Per sapere se un Servizio AWS programma rientra nell'ambito di specifici programmi di conformità, consulta Servizi AWS la sezione [Scope by Compliance Program Servizi AWS](#) e scegli il programma di conformità che ti interessa. Per informazioni generali, consulta Programmi di [AWS conformità Programmi](#) di di .

È possibile scaricare report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta [Scaricamento dei report in AWS Artifact](#) .

La vostra responsabilità di conformità durante l'utilizzo Servizi AWS è determinata dalla sensibilità dei dati, dagli obiettivi di conformità dell'azienda e dalle leggi e dai regolamenti applicabili. AWS fornisce le seguenti risorse per contribuire alla conformità:

- [Guide introduttive su sicurezza e conformità](#): queste guide all'implementazione illustrano considerazioni sull'architettura e forniscono passaggi per implementare ambienti di base incentrati sulla AWS sicurezza e la conformità.
- [Progettazione per la sicurezza e la conformità HIPAA su Amazon Web Services](#): questo white paper descrive in che modo le aziende possono utilizzare AWS per creare applicazioni idonee all'HIPAA.

Note

Non Servizi AWS tutte sono idonee all'HIPAA. Per ulteriori informazioni, consulta la sezione [Riferimenti sui servizi conformi ai requisiti HIPAA](#).

- [AWS Risorse per la per la conformità](#): questa raccolta di cartelle di lavoro e guide potrebbe essere valida per il tuo settore e la tua località.

- [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)).
- [Evaluating Resources with Rules](#) nella AWS Config Developer Guide: il AWS Config servizio valuta la conformità delle configurazioni delle risorse alle pratiche interne, alle linee guida e alle normative del settore.
- [AWS Security Hub](#)— Ciò Servizio AWS fornisce una visione completa dello stato di sicurezza interno. 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](#).
- [Amazon GuardDuty](#): Servizio AWS rileva potenziali minacce ai tuoi carichi di lavoro Account AWS, ai contenitori e ai dati monitorando l'ambiente alla ricerca di attività sospette e dannose. GuardDuty può aiutarti a soddisfare vari requisiti di conformità, come lo standard PCI DSS, soddisfacendo i requisiti di rilevamento delle intrusioni imposti da determinati framework di conformità.
- [AWS Audit Manager](#)— Ciò Servizio AWS consente di verificare continuamente l' AWS utilizzo per semplificare la gestione del rischio e la conformità alle normative e agli standard di settore.

Resilienza in Amazon Kinesis Video Streams

L'infrastruttura AWS globale è costruita attorno AWS a regioni e zone di disponibilità. AWS Le regioni forniscono più zone di disponibilità fisicamente separate e isolate, collegate con reti a bassa latenza, ad alto throughput e altamente ridondanti. Con le zone di disponibilità, è possibile progettare e gestire 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 tradizionali a data center singolo o multiplo.

[Per ulteriori informazioni su AWS regioni e zone di disponibilità, consulta Global Infrastructure.AWS](#)

Sicurezza dell'infrastruttura in Kinesis Video Streams

In quanto servizio gestito, Amazon Kinesis Video Streams è protetto AWS dalle procedure di sicurezza di rete globali descritte [nel white paper Amazon Web Services: Overview of Security Processes](#).

Utilizzi chiamate API AWS pubblicate per accedere a Kinesis Video Streams attraverso la rete. I client devono supportare Transport Layer Security (TLS) 1.2 o versioni successive. I client devono, inoltre, supportare le suite di crittografia 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 di chiave di accesso e una chiave di accesso segreta associata a un principale IAM. O puoi utilizzare [AWS Security Token Service](#) (AWS STS) per generare credenziali di sicurezza temporanee per sottoscrivere le richieste.

Best practice di sicurezza per Kinesis Video Streams

Amazon Kinesis Video Streams offre una serie di funzionalità di sicurezza da prendere in considerazione durante lo sviluppo e l'implementazione delle proprie politiche di sicurezza. Le seguenti best practice sono linee guida generali e non rappresentano una soluzione di sicurezza completa. Poiché queste best practice potrebbero non essere appropriate o sufficienti per l'ambiente, gestiscile come considerazioni utili anziché prescrizioni.

Per le best practice di sicurezza per i tuoi dispositivi remoti, consulta [Best practice per la sicurezza degli agenti dei dispositivi](#).

Implementazione dell'accesso con privilegi minimi

Quando concedi le autorizzazioni, sei tu a decidere chi ottiene quali autorizzazioni per quali risorse Kinesis Video Streams. È possibile abilitare operazioni specifiche che si desidera consentire su tali risorse. Pertanto è necessario concedere solo le autorizzazioni necessarie per eseguire un'attività. L'implementazione dell'accesso con privilegi minimi è fondamentale per ridurre i rischi di sicurezza e l'impatto risultante da errori o intenzioni dannose.

Ad esempio, un produttore che invia dati a Kinesis Video Streams PutMedia richiede GetStreamingEndpoint solo, e. DescribeStream Non concedere autorizzazioni delle applicazioni del producer per tutte le operazioni (*) o per altre operazioni, ad esempio GetMedia.

Per ulteriori informazioni, consulta l'argomento che descrive [qual è il privilegio minimo e perché è necessario](#).

Uso di ruoli IAM

Le applicazioni Producer e Client devono disporre di credenziali valide per accedere a Kinesis Video Streams. Non devi archiviare credenziali AWS direttamente in un'applicazione client o in un bucket

Amazon S3. Si tratta di credenziali a lungo termine che non vengono ruotate automaticamente e che, se compromesse, potrebbero avere un impatto aziendale significativo.

Dovresti invece utilizzare un ruolo IAM per gestire le credenziali temporanee per le tue applicazioni di produzione e client per accedere a Kinesis Video Streams. Quando utilizzi un ruolo, non devi usare credenziali a lungo termine (come nome utente e password o chiavi di accesso) per accedere ad altre risorse.

Per ulteriori informazioni, consulta gli argomenti seguenti nella Guida per l'utente IAM:

- [Ruoli IAM](#)
- [Scenari comuni per ruoli: utenti, applicazioni e servizi](#)

Utilizzalo CloudTrail per monitorare le chiamate API

Kinesis Video Streams AWS CloudTrail funziona con un servizio che fornisce una registrazione delle azioni intraprese da un utente, da un ruolo Servizio AWS o da un utente in Kinesis Video Streams.

È possibile utilizzare le informazioni raccolte CloudTrail per determinare la richiesta effettuata a Kinesis Video Streams, l'indirizzo IP da cui è stata effettuata la richiesta, chi ha effettuato la richiesta, quando è stata effettuata e ulteriori dettagli.

Per ulteriori informazioni, consulta [the section called “Registrazione delle chiamate API di CloudTrail con”](#).

Librerie per produttori di Kinesis Video Streams

Le librerie Amazon Kinesis Video Streams Producer sono un insieme di librerie nel Kinesis Video Streams Producer SDK. Il client utilizza le librerie e l'SDK per creare l'applicazione sul dispositivo per la connessione sicura a Kinesis Video Streams e lo streaming di dati multimediali da visualizzare nella console o nelle applicazioni client in tempo reale.

I dati multimediali possono essere diffusi nei seguenti modi:

- In tempo reale
- Dopo averlo bufferizzato per alcuni secondi
- Dopo il caricamento dei file multimediali

Dopo aver creato uno stream Kinesis Video Streams, puoi iniziare a inviargli dati. È possibile utilizzare l'SDK per creare un codice applicativo che estrae i dati video, noti come frame, dalla sorgente multimediale e li carica su Kinesis Video Streams. Queste applicazioni vengono anche definite applicazioni producer.

Le Producer Library contengono i seguenti componenti:

- [Cliente di Kinesis Video Streams Producer](#)
- [Libreria dei produttori di Kinesis Video Streams](#)

Cliente di Kinesis Video Streams Producer

Il Kinesis Video Streams Producer Client include un `KinesisVideoClient` classe. Questa classe gestisce le fonti multimediali, riceve i dati dalle fonti e gestisce il ciclo di vita dello stream man mano che i dati fluiscono da una fonte multimediale a Kinesis Video Streams. Fornisce inoltre un `MediaSource` interfaccia per definire l'interazione tra Kinesis Video Streams e hardware e software proprietari.

Un'origine multimediale può essere costituita da pressoché qualsiasi elemento. Ad esempio, è possibile utilizzare un'origine multimediale sotto forma di fotocamera o microfono. Le origini multimediali sorgenti non si limitano a sorgenti audio e video. Ad esempio, i log dei dati potrebbero essere file di testo, che tuttavia possono sempre essere inviati come flusso di dati. Puoi inoltre disporre di più telecamere sul tuo telefono che trasmettono in streaming i dati simultaneamente.

Per ottenere i dati da qualsiasi di queste origini, puoi implementare l'interfaccia `MediaSource`. Questa interfaccia consente ulteriori scenari per cui non forniamo il supporto integrato. Ad esempio, potresti scegliere di inviare quanto segue a Kinesis Video Streams:

- Un flusso di dati diagnostici (ad esempio, i log dell'applicazione e gli eventi)
- I dati provenienti da telecamere a infrarossi, RADAR o telecamere di profondità

Kinesis Video Streams non fornisce implementazioni integrate per dispositivi di produzione multimediale come le fotocamere. Per estrarre i dati provenienti da questi dispositivi, è necessario implementare il codice, creando la tua implementazione di origine multimediale personalizzate. Puoi quindi registrare in modo esplicito le tue fonti multimediali personalizzate con `KinesisVideoClient`, che carica i dati su Kinesis Video Streams.

Il Kinesis Video Streams Producer Client è disponibile per applicazioni Java e Android. Per ulteriori informazioni, consultare [Utilizzo di Java Producer Library](#) e [Utilizzo di Android Producer Library](#).

Libreria dei produttori di Kinesis Video Streams

La Kinesis Video Streams Producer Library è contenuta nel Kinesis Video Streams Producer Client. La libreria può anche essere utilizzata direttamente per coloro che desiderano un'integrazione più profonda con Kinesis Video Streams. Ciò consente l'integrazione da dispositivi con sistemi operativi proprietari, stack di rete oppure risorse sul dispositivo limitate.

La Kinesis Video Streams Producer Library implementa la macchina a stati per lo streaming su Kinesis Video Streams. Offre hook di callback, che ti richiedono di fornire la tua implementazione di trasporto e di gestire in modo esplicito ogni messaggio da e verso il servizio.

È possibile scegliere di utilizzare direttamente la Kinesis Video Streams Producer Library per i seguenti motivi:

- Il dispositivo in cui vuoi eseguire l'applicazione non ha una macchina virtuale Java.
- Vuoi scrivere il codice dell'applicazione in altri linguaggi diversi da Java.
- Vuoi ridurre la quantità di sovraccarico del codice e limitarlo al minimo livello di astrazione, a causa di limitazioni come la memoria e la potenza di elaborazione.

Attualmente, la Kinesis Video Streams Producer Library è disponibile per applicazioni Android, C, C++ e Java. Per ulteriori informazioni, consulta le lingue supportate di seguito [Argomenti correlati](#).

Argomenti correlati

[Utilizzo di Java Producer Library](#)

[Utilizzo di Android Producer Library](#)

[Utilizzo di C++ Producer Library](#)

[Utilizzo di C Producer Library](#)

[Uso dell'SDK del producer C++ su Raspberry Pi](#)

Utilizzo di Java Producer Library

Puoi utilizzare la Java Producer Library fornita da Amazon Kinesis Video Streams per scrivere codice applicativo con una configurazione minima, per inviare dati multimediali da un dispositivo a un flusso video Kinesis.

Esegui i seguenti passaggi per integrare il codice con Kinesis Video Streams in modo che l'applicazione possa avviare lo streaming di dati sullo streaming video Kinesis:

1. Crea un'istanza dell'oggetto `KinesisVideoClient`.
2. Crea un oggetto `MediaSource` fornendo informazioni sull'origine multimediale. Ad esempio, durante la creazione di un'origine multimediale fotocamera, devi fornire informazioni quali l'identificazione della fotocamera e specificare la codifica utilizzata dalla fotocamera.

Quando vuoi avviare lo streaming, devi creare un'origine multimediale personalizzata.

3. Registra l'origine multimediale con `KinesisVideoClient`.

Dopo che hai registrato l'origine multimediale con `KinesisVideoClient`, ogni volta che i dati diventano disponibili con tale origine, viene chiamato `KinesisVideoClient` con i dati.

Procedura: utilizzo dell'SDK Java Producer

Questa procedura dimostra come utilizzare il Kinesis Video Streams Java Producer Client nell'applicazione Java per inviare dati al flusso video Kinesis.

Per questa procedura non è necessario disporre di un'origine multimediale, come una videocamera o un microfono. Al contrario, a scopo di test, il codice genera frame di esempio composti da una serie di

byte. Puoi utilizzare lo stesso modello di codifica quando invii dati multimediali da origini reali, come videocamere e microfoni.

La procedura include le seguenti fasi:

- [Download e configurazione del codice](#)
- [Scrittura e analisi del codice](#)
- [Esecuzione e verifica del codice](#)

Prerequisiti

- Nel codice di esempio, fornisci le credenziali specificando un profilo che hai impostato nel file di profilo delle credenziali. AWS Se non lo hai già fatto, configura prima il profilo delle credenziali. Per ulteriori informazioni, consulta [Configurare AWS le credenziali e la regione per](#) lo sviluppo in. AWS SDK for Java

Note

L'esempio Java utilizza un `SystemPropertiesCredentialsProvider` oggetto per ottenere le credenziali dell'utente. Il provider recupera queste credenziali dalle proprietà del sistema Java `aws.accessKeyId` e `aws.secretKey`. Puoi impostare queste proprietà nel tuo ambiente di sviluppo Java. Per informazioni su come impostare le proprietà del sistema Java, consulta la documentazione per il tuo ambiente di sviluppo integrato (IDE) specifico.

- Il file `NativeLibraryPath` deve contenere il `KinesisVideoProducerJNI` file, disponibile all'[indirizzo https://github.com/aws-labs/amazon-kinesis-video-streams - producer-sdk-cpp](https://github.com/aws-labs/amazon-kinesis-video-streams-producer-sdk-cpp).

L'estensione del nome di questo file varia a seconda del sistema operativo:

- `KinesisVideoProducerJNI.so` per Linux
- `KinesisVideoProducerJni.dylib` per macOS
- `KinesisVideoProducerJNI.dll` per Windows

Note

Le librerie predefinite per macOS, Ubuntu, Windows e Raspbian sono disponibili all'[indirizzo amazon-kinesis-video-streams https://github.com/aws-labs/ - src/main/resources/lib .git. producer-sdk-java](https://github.com/aws-labs/amazon-kinesis-video-streams-producer-sdk-java) Per altri ambienti, compila [C++ Producer Library](#).

Passaggio 1: scaricare e configurare il codice della Java Producer Library

In questa sezione della procedura per Java Producer Library puoi scaricare il codice di esempio Java, importare il progetto nell'IDE Java e configurare i percorsi della libreria.

Per i prerequisiti e altri dettagli su questo esempio, consulta l'articolo sull'[utilizzo di Java Producer Library](#).

1. Create una directory, quindi clonate il codice sorgente di esempio dal GitHub repository.

```
git clone https://github.com/awslabs/amazon-kinesis-video-streams-producer-sdk-java
```

2. Apri l'ambiente di sviluppo integrato (IDE) Java che utilizzi (ad esempio, [Eclipse](#) o [JetBrains IntelliJ IDEA](#)) e importa il progetto Apache Maven che hai scaricato:
 - In IntelliJ IDEA: scegli Import (Importa). Andare al file pom.xml nella radice del pacchetto scaricato.
 - In Eclipse: scegliere File, Import (Importa), Maven, Existing Maven Projects (Progetti Maven esistenti). Passa quindi alla directory kinesis-video-java-demo.

Per ulteriori informazioni, consulta la documentazione relativa al tuo IDE.

3. Il codice di esempio Java utilizza le credenziali correnti. AWS Per usare un profilo di credenziali differenti, individua il seguente codice in DemoAppMain.java:

```
final KinesisVideoClient kinesisVideoClient = KinesisVideoJavaClientFactory
    .createKinesisVideoClient(
        Regions.US_WEST_2,
        AuthHelper.getSystemPropertiesCredentialsProvider());
```

Modifica il codice come indicato di seguito:

```
final KinesisVideoClient kinesisVideoClient = KinesisVideoJavaClientFactory
    .createKinesisVideoClient(
        Regions.US_WEST_2,
        new ProfileCredentialsProvider("credentials-profile-name"));
```

Per ulteriori informazioni, vedere [ProfileCredentialsProvider](#) nel AWS SDK for Java riferimento.

Fase successiva

[the section called “Fase 2: Scrivere ed esaminare il codice”](#)

Fase 2: Scrivere ed esaminare il codice

In questa sezione della [procedura Java Producer Library](#), si scrive ed si esamina il codice di esempio Java scaricato nella sezione precedente.

L'applicazione per i test di Java ([DemoAppMain](#)) mostra il seguente modello di codifica:

- Creare un'istanza di `KinesisVideoClient`.
- Creare un'istanza di `MediaSource`.
- Registra `MediaSource` con il client.
- Avvia lo streaming. Avviate il `MediaSource` e inizia a inviare i dati al client.

Nelle seguenti sezioni sono fornite maggiori informazioni.

Creazione di un'istanza di `KinesisVideoClient`

Per creare l'oggetto `KinesisVideoClient`, richiama l'operazione `createKinesisVideoClient`.

```
final KinesisVideoClient kinesisVideoClient = KinesisVideoJavaClientFactory
    .createKinesisVideoClient(
        Regions.US_WEST_2,
        AuthHelper.getSystemPropertiesCredentialsProvider());
```

Per eseguire chiamate alla rete, `KinesisVideoClient` ha bisogno di credenziali da autenticare. Puoi trasferire un'istanza di `SystemPropertiesCredentialsProvider`, che legge `AWSCredentials` per il profilo predefinito nel file di credenziali:

```
[default]
aws_access_key_id = ABCDEFGHIJKLMOPQRSTU
aws_secret_access_key = AbCd1234EfGh5678IjKl9012MnOp3456QrSt7890
```

Creazione di un'istanza di MediaSource

Per inviare byte al flusso video Kinesis, devi produrre i dati. Amazon Kinesis Video Streams MediaSource fornisce l'interfaccia che rappresenta l'origine dei dati.

Ad esempio, la libreria Java Kinesis Video Streams ImageFileMediaSource fornisce l'implementazione MediaSource dell'interfaccia. Questa classe legge solo i dati da una serie di file multimediali anziché da un flusso video Kinesis, ma puoi usarla per testare il codice.

```
final MediaSource bytesMediaSource = createImageFileMediaSource();
```

Registrazione del con il client MediaSource

Registra l'origine multimediale creata con KinesisVideoClient in modo che sia a conoscenza dell'esistenza del client e possa quindi inviargli i dati.

```
kinesisVideoClient.registerMediaSource(mediaSource);
```

Avvio della sorgente multimediale

Avviate la sorgente multimediale in modo che possa iniziare a generare dati e inviarli al client.

```
bytesMediaSource.start();
```

Approfondimenti

[the section called "Passaggio 3: Esegui e verifica il codice"](#)

Passaggio 3: Esegui e verifica il codice

Per eseguire il test harness Java per la [Java Producer Library](#), effettuate le seguenti operazioni.

1. Scegliete DemoAppMain.
2. Scegli Esegui, Esegui 'DemoAppMain'.
3. Aggiungi le tue credenziali per gli argomenti JVM per l'applicazione:
 - Per AWS credenziali non temporanee: "-Daws.accessKeyId={YourAwsAccessKey} -Daws.secretKey={YourAwsSecretKey} -Djava.library.path={NativeLibraryPath}"

- Per le credenziali temporanee AWS : "-Daws.accessKeyId={YourAwsAccessKey} -Daws.secretKey={YourAwsSecretKey} -Daws.sessionToken={YourAwsSessionToken} -Djava.library.path={NativeLibraryPath}"

4. Accedi AWS Management Console e apri la console [Kinesis Video Streams](#).

Nella pagina Manage Streams (Gestisci flussi), scegli il tuo flusso.

5. Il video di esempio viene riprodotto nel lettore integrato. La visualizzazione del video potrebbe richiedere una breve attesa (fino a 10 secondi con la larghezza di banda e le condizioni del processore abituali) affinché vengano caricati i fotogrammi.

L'esempio di codice crea un flusso. Appena `MediaSource` nel codice si avvia, inizia a inviare frame di esempio a `KinesisVideoClient`. Il client invia quindi i dati al flusso video Kinesis.

Utilizzo di Android Producer Library

Puoi utilizzare la Android Producer Library fornita da Amazon Kinesis Video Streams per scrivere codice applicativo, con una configurazione minima, per inviare dati multimediali da un dispositivo Android a un flusso video Kinesis.

Esegui i seguenti passaggi per integrare il codice con Kinesis Video Streams in modo che l'applicazione possa avviare lo streaming di dati sullo streaming video Kinesis:

1. Crea un'istanza dell'oggetto `KinesisVideoClient`.
2. Crea un oggetto `MediaSource` fornendo informazioni sull'origine multimediale. Ad esempio, durante la creazione di un'origine multimediale fotocamera, devi fornire informazioni quali l'identificazione della fotocamera e specificare la codifica utilizzata dalla fotocamera.

Quando vuoi avviare lo streaming, devi creare un'origine multimediale personalizzata.

Procedura: utilizzo dell'SDK del producer Android

Questa procedura illustra come utilizzare il Kinesis Video Streams Android Producer Client nell'applicazione Android per inviare dati al flusso video Kinesis.

La procedura include le seguenti fasi:

- [the section called “Prerequisiti”](#)
- [the section called “Fase 1: download e configurazione del codice”](#)
- [the section called “Fase 2: Esamina il codice”](#)
- [the section called “Passaggio 3: Esegui e verifica il codice”](#)

Prerequisiti

- Ti consigliamo di utilizzare [Android Studio](#) per esaminare, modificare ed eseguire il codice dell'applicazione. Ti consigliamo di utilizzare l'ultima versione stabile.
- Nel codice di esempio, fornisci le credenziali di Amazon Cognito.

Segui queste procedure per configurare un pool di utenti e un pool di identità di Amazon Cognito.

- [Configura un pool di utenti](#)
- [Configura un pool di identità](#)

Configura un pool di utenti

Per configurare un pool di utenti

1. Accedi alla [console Amazon Cognito](#) e verifica che la regione sia corretta.
2. Nella barra di navigazione a sinistra, scegli Pool di utenti.
3. Nella sezione Pool di utenti, scegli Crea pool di utenti.
4. Completa le seguenti sezioni:
 - a. Passaggio 1: Configurazione dell'esperienza di accesso - Nella sezione delle opzioni di accesso al pool di utenti di Cognito, seleziona le opzioni appropriate.
Seleziona Avanti.
 - b. Passaggio 2: Configurazione dei requisiti di sicurezza: seleziona le opzioni appropriate.
Seleziona Avanti.
 - c. Passaggio 3: Configura l'esperienza di registrazione: seleziona le opzioni appropriate.
Seleziona Avanti.
 - d. Fase 4: Configurazione del recapito dei messaggi - Seleziona le opzioni appropriate.

Nel campo di selezione del ruolo IAM, seleziona un ruolo esistente o crea un nuovo ruolo.

Seleziona Avanti.

- e. Passaggio 5: Integra la tua app: seleziona le opzioni appropriate.

Nel campo Client iniziale dell'app, scegli Client riservato.

Seleziona Avanti.

- f. Passaggio 6: Rivedi e crea: rivedi le selezioni effettuate nelle sezioni precedenti, quindi scegli Crea pool di utenti.

5. Nella pagina Pool di utenti, seleziona il pool che hai appena creato.

Copia l'ID del pool di utenti e prendine nota per dopo. Nel `awsconfiguration.json` file, questo è `CognitoUserPool.Default.PoolId`.

6. Seleziona la scheda Integrazione dell'app e vai in fondo alla pagina.

7. Nella sezione Elenco dei client dell'app, scegli il nome del client dell'app che hai appena creato.

Copia l'ID cliente e prendine nota per dopo. Nel `awsconfiguration.json` file, questo è `CognitoUserPool.Default.AppClientId`.

8. Mostra il segreto del Cliente e prendine nota per dopo. Nel `awsconfiguration.json` file, questo è `CognitoUserPool.Default.AppClientSecret`.

Configura un pool di identità

Per configurare un pool di identità

1. Accedi alla [console Amazon Cognito](#) e verifica che la regione sia corretta.
2. Nella barra di navigazione a sinistra, scegli Identity pool.
3. Scegli Crea pool di identità.
4. Configura il pool di identità.
 - a. Passaggio 1: Configurare l'affidabilità del pool di identità: completare le seguenti sezioni:
 - Accesso utente: seleziona Accesso autenticato
 - Fonti di identità autenticate: seleziona il pool di utenti Amazon Cognito

Seleziona Avanti.

- b. Fase 2: Configurazione delle autorizzazioni - Nella sezione Ruolo autenticato, completa i seguenti campi:

- Ruolo IAM: seleziona Crea un nuovo ruolo IAM
- Nome del ruolo IAM: inserisci un nome e prendine nota per un passaggio successivo.

Seleziona Avanti.

- c. Passaggio 3: Connect identity provider - Nella sezione Dettagli del pool di utenti completa i seguenti campi:

- ID del pool di utenti: seleziona il pool di utenti creato in precedenza.
- ID client dell'app: seleziona l'ID client dell'app che hai creato in precedenza.

Seleziona Avanti.

- d. Passaggio 4: Configurazione delle proprietà: digitare un nome nel campo Nome del pool di identità.

Seleziona Avanti.

- e. Passaggio 5: Rivedi e crea: rivedi le selezioni in ciascuna delle sezioni, quindi seleziona Crea pool di identità.

5. Nella pagina Pool di identità, seleziona il tuo nuovo pool di identità.

Copia l'ID del pool di identità e prendine nota per dopo. Nel `awsconfiguration.json` file, questo è `CredentialsProvider.CognitoIdentity.Default.PoolId`.

6. Aggiorna le autorizzazioni per il ruolo IAM.

- Accedi AWS Management Console e apri la console IAM all'[indirizzo https://console.aws.amazon.com/iam/](https://console.aws.amazon.com/iam/).
- Nella barra di navigazione a sinistra, scegli Ruoli.
- Trova e seleziona il ruolo che hai creato sopra.

Note

Usa la barra di ricerca, se necessario.

- d. Seleziona la politica di autorizzazioni allegata.

Seleziona Edit (Modifica).

- e. Seleziona la scheda JSON e sostituisci la politica con la seguente:

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

Seleziona Avanti.

- f. Seleziona la casella accanto a Imposta questa nuova versione come predefinita se non è già selezionata.

Seleziona Salva modifiche.

Passaggio 1: scarica e configura il codice di Android Producer Library

In questa sezione della procedura per Android Producer Library, puoi scaricare il codice di esempio per Android e aprire il progetto in Android Studio.

Per i prerequisiti e altri dettagli su questo esempio, consulta [Utilizzo di Android Producer Library](#).

1. Create una directory, quindi clonata AWS Mobile SDK for Android dal GitHub repository.

```
git clone https://github.com/aws-labs/aws-sdk-android-samples
```

2. Apri [Android Studio](#).
3. Nella schermata iniziale, scegli Apri un progetto di Android Studio esistente.
4. Passa alla directory `aws-sdk-android-samples/AmazonKinesisVideoDemoApp` e scegli OK.
5. Apri il file `AmazonKinesisVideoDemoApp/src/main/res/raw/awsconfiguration.json`.

Nel `CredentialsProvider` nodo, fornisci l'ID del pool di identità indicato nella procedura Per impostare un pool di identità nella sezione [Prerequisiti](#) e fornisci il tuo Regione AWS (ad esempio,). **us-west-2**

Nel `CognitoUserPool` nodo, fornisci il segreto del client dell'app, l'ID client dell'app e l'ID del pool dalla procedura Per impostare un pool di utenti nella sezione [Prerequisiti](#) e fornisci i tuoi Regione AWS (ad esempio, **us-west-2**).

6. Il file `awsconfiguration.json` sarà simile a quanto riportato di seguito:

```
{
  "Version": "1.0",
  "CredentialsProvider": {
    "CognitoIdentity": {
      "Default": {
        "PoolId": "us-west-2:01234567-89ab-cdef-0123-456789abcdef",
        "Region": "us-west-2"
      }
    }
  },
  "IdentityManager": {
    "Default": {}
  },
  "CognitoUserPool": {
    "Default": {
      "AppClientSecret": "abcdefghijklmnopqrstuvwxy0123456789abcdefghijklmnop",
      "AppClientId": "0123456789abcdefghijklmnop",
      "PoolId": "us-west-2_qRsTuVwXy",
      "Region": "us-west-2"
    }
  }
}
```

```
}
```

7. Aggiorna il file `AmazonKinesisVideoDemoApp/src/main/java/com/amazonaws/kinesisvideo/demoapp/KinesisVideoDemoApp.java` con la tua regione (nell'esempio seguente, è impostato su `US_WEST_2`):

```
public class KinesisVideoDemoApp extends Application {  
    public static final String TAG = KinesisVideoDemoApp.class.getSimpleName();  
    public static Regions KINESIS_VIDEO_REGION = Regions.US_WEST_2;  
}
```

[Per informazioni sulle Regione AWS costanti, vedi Regioni.](#)

Approfondimenti

[the section called “Fase 2: Esamina il codice”](#)

Fase 2: Esamina il codice

In questa sezione della [procedura per Android Producer Library](#), puoi esaminare il codice di esempio.

L'applicazione per i test di Android (`AmazonKinesisVideoDemoApp`) mostra il seguente modello di codifica:

- Creare un'istanza di `KinesisVideoClient`.
- Creare un'istanza di `MediaSource`.
- Avvia lo streaming. Avvia il `MediaSource`, e inizia a inviare i dati al client.

Nelle seguenti sezioni sono fornite maggiori informazioni.

Creazione di un'istanza di `KinesisVideoClient`

Per creare l'oggetto [KinesisVideoClient](#), richiama l'operazione [createKinesisVideoClient](#).

```
mKinesisVideoClient = KinesisVideoAndroidClientFactory.createKinesisVideoClient(  
    getActivity(),  
    KinesisVideoDemoApp.KINESIS_VIDEO_REGION,  
    KinesisVideoDemoApp.getCredentialsProvider());
```

Per eseguire chiamate alla rete, `KinesisVideoClient` ha bisogno di credenziali da autenticare. Passi un'istanza di `AWSCredentialsProvider`, che legge le tue credenziali di Amazon Cognito `awsconfiguration.json` dal file che hai modificato nella sezione precedente.

Creazione di un'istanza di `MediaSource`

Per inviare byte al flusso video Kinesis, devi produrre i dati. Amazon Kinesis Video Streams [MediaSource](#) fornisce l'interfaccia che rappresenta l'origine dei dati.

Ad esempio, la libreria Android Kinesis Video Streams [AndroidCameraMediaSource](#) fornisce l'implementazione `MediaSource` dell'interfaccia. Questa classe legge i dati da una delle fotocamere del dispositivo.

Nel seguente esempio di codice (dal file [fragment/StreamConfigurationFragment.java](#)), viene creata la configurazione per l'origine multimediale:

```
private AndroidCameraMediaSourceConfiguration getCurrentConfiguration() {
    return new AndroidCameraMediaSourceConfiguration(
        AndroidCameraMediaSourceConfiguration.builder()
            .withCameraId(mCamerasDropdown.getSelectedItem().getCameraId())

            .withEncodingMimeType(mMimeTypeDropdown.getSelectedItem().getMimeType())

            .withHorizontalResolution(mResolutionDropdown.getSelectedItem().getWidth())

            .withVerticalResolution(mResolutionDropdown.getSelectedItem().getHeight())
                .withCameraFacing(mCamerasDropdown.getSelectedItem().getCameraFacing())
                .withIsEncoderHardwareAccelerated(
                    mCamerasDropdown.getSelectedItem().isEncoderHardwareAccelerated())
                .withFrameRate(FRAMERATE_20)
                .withRetentionPeriodInHours(RETENTION_PERIOD_48_HOURS)
                .withEncodingBitRate(BITRATE_384_KBPS)
                .withCameraOrientation(-
                    mCamerasDropdown.getSelectedItem().getCameraOrientation())

            .withNalAdaptationFlags(StreamInfo.NalAdaptationFlags.NAL_ADAPTATION_ANNEXB_CPD_AND_FRAME_NALS)
                .withIsAbsoluteTimecode(false));
}
```

Nel seguente esempio di codice (dal file [fragment/StreamingFragment.java](#)), viene creata la configurazione per l'origine multimediale:

```
mCameraMediaSource = (AndroidCameraMediaSource) mKinesisVideoClient
    .createMediaSource(mStreamName, mConfiguration);
```

Avvio della sorgente multimediale

Avvia l'origine multimediale in modo che inizi a generare dati e a inviarli al client. I seguenti esempi di codice sono tratti dal file [fragment/StreamingFragment.java](#):

```
mCameraMediaSource.start();
```

Approfondimenti

[the section called “Passaggio 3: Esegui e verifica il codice”](#)

Passaggio 3: Esegui e verifica il codice

Per eseguire l'applicazione di esempio Android per [Android Producer Library](#), esegui le operazioni descritte di seguito.

1. Esegui la connessione a un dispositivo Android.
2. Scegli Run (Esegui), Run... (Esegui...), quindi scegli Edit configurations... (Modifica configurazioni...).
3. Scegli l'icona più (+), App Android. Nel campo Name (Nome), inserire **AmazonKinesisVideoDemoApp**. Nel menu a discesa Modulo, scegli AmazonKinesisVideoDemoApp. Scegli OK.
4. Scegli Run (Esegui), Run (Esegui).
5. Nella schermata Select Deployment Target (Seleziona destinazione distribuzione) scegli il dispositivo connesso, quindi scegli OK.
6. Nell'AWSKinesisVideoDemoApp applicazione sul dispositivo, scegli Crea nuovo account.
7. Inserisci i valori per USERNAME (NOME UTENTE), Password, Given name (Nome specificato), Email address (Indirizzo e-mail) e Phone number (Numero di telefono), quindi scegli Sign up (Iscrizione).

Note

Questi valori hanno le seguenti limitazioni:

- Password: deve contenere lettere maiuscole e minuscole, numeri e caratteri speciali. Puoi modificare questi vincoli nella pagina del pool di utenti sulla console [Amazon Cognito](#).
- Email address (Indirizzo e-mail): deve essere un indirizzo valido a cui ricevere un codice di conferma.
- Phone number (Numero di telefono): deve avere il seguente formato: **+<Country code><Number>**, ad esempio, **+12065551212** .

8. Inserisci il codice che ricevi via e-mail e scegli Conferma. Scegli OK.
9. Nella pagina successiva, mantieni i valori predefiniti e scegli Stream.
10. Accedere AWS Management Console e aprire la console [Kinesis Video Streams](#) nella regione Stati Uniti occidentali (Oregon).

Nella pagina Manage Streams (Gestisci flussi), scegli demo-stream.

11. Il video streaming viene riprodotto nel lettore integrato. La visualizzazione del video potrebbe richiedere una breve attesa (fino a 10 secondi con la larghezza di banda e le condizioni del processore abituali) affinché vengano caricati i fotogrammi.

Note

Se lo schermo del dispositivo ruota (ad esempio, da verticale a orizzontale), l'applicazione interrompe lo streaming del video.

L'esempio di codice crea un flusso. Appena `MediaSource` nel codice viene avviato, inizia a inviare fotogrammi dalla fotocamera a `KinesisVideoClient`. Il client invia quindi i dati a un flusso video Kinesis denominato demo-stream.

Utilizzo di C++ Producer Library

Puoi utilizzare la libreria C++ Producer fornita da Amazon Kinesis Video Streams per scrivere codice applicativo per inviare dati multimediali da un dispositivo a un flusso video Kinesis.

Modello a oggetti

La libreria C++ fornisce i seguenti oggetti per gestire l'invio di dati a un flusso video Kinesis:

- `KinesisVideoProducer`: contiene informazioni sulla fonte multimediale e sulle AWS credenziali e mantiene i callback per segnalare gli eventi di Kinesis Video Streams.
- `KinesisVideoStream`: rappresenta il flusso video di Kinesis. Contiene informazioni sui parametri del flusso video, come il nome, il periodo di conservazione dei dati e il tipo di contenuto multimediale.

Inserimento di contenuti multimediali nello stream

È possibile utilizzare i metodi forniti dalla libreria C++ (ad esempio, `PutFrame`) per inserire dati nell'`KinesisVideoStream` oggetto. La libreria gestisce quindi lo stato interno dei dati, che può includere le seguenti attività:

- Esecuzione dell'autenticazione.
- Controllo della latenza di rete. Se la latenza è troppo elevata, la libreria potrebbe scegliere di eliminare i fotogrammi.
- Monitoraggio dello stato dello streaming in corso.

Interfacce di callback

Questo livello mostra un set di interfacce di richiamata, che consentono la comunicazione con il livello dell'applicazione. Queste interfacce di richiamata includono:

- `Service callbacks interface (CallbackProvider)`: la libreria richiama gli eventi ottenuti tramite questa interfaccia quando crea uno stream, ottiene una descrizione dello stream ed elimina uno stream.
- `Interfaccia con stato client pronto o eventi di storage ridotto (ClientCallbackProvider)`: la libreria richiama gli eventi su questa interfaccia quando il client è pronto oppure quando rileva che potrebbe esaurire lo storage o la memoria disponibile.
- `Interfaccia di richiamata degli eventi del flusso (StreamCallbackProvider)`: la libreria richiama gli eventi su questa interfaccia quando si verificano eventi del flusso, ad esempio quando lo stato del flusso diventa pronto, quando i fotogrammi vengono saltati o in caso di errori del flusso.

Kinesis Video Streams fornisce implementazioni predefinite per queste interfacce. Puoi anche fornire un'implementazione personalizzata, ad esempio, se hai bisogno di una logica di rete personalizzata o desideri esporre all'interfaccia utente una condizione di archiviazione ridotta.

Per ulteriori informazioni sui callback in Producer Library, consulta [Richiamate SDK di Producer](#).

Procedura: utilizzo dell'SDK del producer C++

Questa procedura dimostra come utilizzare il client Kinesis Video Streams e le sorgenti multimediali in un'applicazione C++ per inviare dati al flusso video Kinesis.

La procedura include le seguenti fasi:

- [Fase 1: download e configurazione del codice](#)
- [Fase 2: scrittura e analisi del codice](#)
- [Fase 3: esecuzione e verifica del codice](#)

Prerequisiti

- **Credenziali:** nel codice di esempio, si forniscono le credenziali specificando un profilo impostato nel file di profilo delle credenziali. AWS Se non lo hai già fatto, configura prima il profilo delle credenziali.

Per ulteriori informazioni, consulta [Configurare le AWS credenziali](#) e la regione per lo sviluppo.

- **Integrazione dell'archivio certificati:** la Kinesis Video Streams Producer Library deve stabilire un rapporto di fiducia con il servizio che chiama. Ciò avviene tramite la convalida delle autorità di certificazione (CA) nell'archivio di certificati pubblico. Sui modelli basati su Linux, questo store si trova nella directory `/etc/ssl/`.

Scarica il certificato dal percorso seguente per lo store dei certificati:

<https://www.amazontrust.com/repository/SFSRootCAG2.pem>

- Installa le seguenti dipendenze di build per macOS:
 - [Autoconf 2.69](#) (Licenza GPLv3+/Autoconf: GNU GPL versione 3 o successive)
 - [CMake 3.7 o 3.8](#)
 - [Pkg-Config](#)
 - xCode (macOS) / clang / gcc (xcode-select versione 2347)
 - Java Development Kit (JDK) (per la compilazione di Java JNI)
 - [Lib-Pkg](#)
- Installa le seguenti dipendenze di build per Ubuntu:

- Git: `sudo apt install git`
- [CMake](#): `sudo apt install cmake`
- G++: `sudo apt install g++`
- configurazione pkg: `sudo apt install pkg-config`
- OpenJDK: `sudo apt install openjdk-8-jdk`

Note

Questo è necessario solo se stai creando Java Native Interface (JNI).

- Imposta la variabile di ambiente JAVA_HOME: `export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/`

Approfondimenti

[Fase 1: download e configurazione del codice di C++ Producer Library](#)

Passaggio 1: scarica e configura il codice C++ Producer Library

Per informazioni su come scaricare e configurare la libreria C++ Producer, consulta [Amazon Kinesis Video Streams CPP Producer, GStreamer Plugin e JNI](#).

[Per i prerequisiti e ulteriori informazioni su questo esempio, consulta Using the C++ Producer Library.](#)

Approfondimenti

[Fase 2: Scrivere ed esaminare il codice](#)

Fase 2: Scrivere ed esaminare il codice

In questa sezione della [procedura per C++ Producer Library](#), potrai esaminare il codice nel test harness C++ (`tst/ProducerTestFixture.h` e altri file). Hai scaricato questo codice nella sezione precedente.

L'esempio di C++ indipendente dalla piattaforma mostra i seguenti modelli di codifica:

- Crea un'istanza di `KinesisVideoProducer` per accedere a Kinesis Video Streams.
- Creare un'istanza di `KinesisVideoStream`. Questo crea un flusso video Kinesis nel tuo Account AWS se non esiste già uno stream con lo stesso nome.

- Chiama `putFrame` in `KinesisVideoStream` per ogni fotogramma di dati, non appena diventa disponibile, per inviarlo al flusso.

Le sezioni seguenti forniscono ulteriori informazioni su questo modello di codifica.

Creazione di un'istanza di `KinesisVideoProducer`

Per creare l'oggetto `KinesisVideoProducer`, richiama il metodo `KinesisVideoProducer::createSync`. L'esempio seguente crea `KinesisVideoProducer` nel file `ProducerTestFixture.h`:

```
kinesis_video_producer_ = KinesisVideoProducer::createSync(move(device_provider_),
    move(client_callback_provider_),
    move(stream_callback_provider_),
    move(credential_provider_),
    defaultRegion_);
```

Il metodo `createSync` accetta i parametri seguenti:

- Un oggetto `DeviceInfoProvider`, che restituisce un oggetto `DeviceInfo` contenente informazioni sul dispositivo o sulla configurazione dello storage.

Note

Puoi configurare la dimensione dello store dei contenuti utilizzando il parametro `deviceInfo.storageInfo.storageSize`. I flussi di contenuti condividono lo store dei contenuti. Per determinare i requisiti delle dimensioni di storage, moltiplica la dimensione media dei fotogrammi per il numero di fotogrammi archiviati per la durata massima di tutti i flussi. Quindi moltiplica per 1,2 per prendere in considerazione la deframmentazione. Supponi, ad esempio, che la tua applicazione abbia la seguente configurazione:

- Tre flussi
- 3 minuti di durata massima
- Ogni flusso è di 30 fotogrammi al secondo (FPS)
- Ogni fotogramma ha una dimensione di 10.000 KB

Il requisito di archiviazione dei contenuti per questa applicazione è $3 \text{ (stream)} * 3 \text{ (minuti)} * 60 \text{ (secondi al minuto)} * 10000 \text{ (kb)} * 1,2 \text{ (margine di deframmentazione)} = 194,4 \text{ Mb} \sim 200 \text{ Mb}$.

- Un oggetto `ClientCallbackProvider`, che restituisce puntatori delle funzioni che segnalano eventi specifici del client.
- Un oggetto `StreamCallbackProvider`, che restituisce puntatori delle funzioni richiamati quando si verificano eventi specifici del flusso.
- Un `CredentialProvider` oggetto che fornisce l'accesso alle variabili di ambiente delle credenziali. AWS
- Il Regione AWS («us-west-2»). L'endpoint del servizio è determinato dalla regione.

Creazione di un'istanza di `KinesisVideoStream`

Per creare l'oggetto `KinesisVideoStream`, richiama il metodo `KinesisVideoProducer::CreateStream` con un parametro `StreamDefinition`. L'esempio crea `KinesisVideoStream` nel file `ProducerTestFixture.h` con il tipo di traccia video e con id traccia pari a 1:

```
auto stream_definition = make_unique<StreamDefinition>(stream_name,
                                                       hours(2),
                                                       tags,
                                                       "",
                                                       STREAMING_TYPE_REALTIME,
                                                       "video/h264",
                                                       milliseconds::zero(),
                                                       seconds(2),
                                                       milliseconds(1),
                                                       true,
                                                       true,
                                                       true);

return kinesis_video_producer_->createStream(move(stream_definition));
```

L'oggetto `StreamDefinition` dispone dei campi seguenti:

- Nome del flusso.
- Periodo di retention dei dati.

- Tag per il flusso. Questi tag possono essere utilizzati dalle applicazioni dei consumatori per trovare il flusso corretto o per avere maggiori informazioni sul flusso. I tag possono anche essere visualizzati in AWS Management Console.
- AWS KMS chiave di crittografia per lo stream. Per ulteriori informazioni, consulta [Utilizzo della crittografia lato server con Kinesis Video Streams](#).
- Tipo di streaming. Attualmente, l'unico valore valido è STREAMING_TYPE_REALTIME.
- Tipo di contenuti multimediali.
- Latenza file multimediali. Questo valore non è attualmente utilizzato e deve essere impostato su 0.
- Durata della riproduzione di ogni frammento.
- Scala del timecode dei file multimediali.
- Se i file multimediali usano la frammentazione del fotogramma chiave.
- Se i file multimediali utilizzano i timecode.
- Se i file multimediali usano i tempi assoluti del frammento.

Aggiungere una traccia audio allo stream video Kinesis

Puoi aggiungere i dettagli della traccia audio a una definizione di flusso di tracce video utilizzando il metodo `addTrack` di: `StreamDefinition`

```
stream_definition->addTrack(DEFAULT_AUDIO_TRACKID, DEFAULT_AUDIO_TRACK_NAME,  
    DEFAULT_AUDIO_CODEC_ID, MKV_TRACK_INFO_TYPE_AUDIO);
```

Il metodo `addTrack` richiede i parametri seguenti:

- ID della traccia (come se si trattasse dell'audio). Questo deve essere unico e un valore diverso da zero.
- Nome della traccia definito dall'utente (ad esempio, «audio» per la traccia audio).
- ID codec per questa traccia (ad esempio, per la traccia audio «A_AAC»).
- Tipo di traccia (ad esempio, utilizzate il valore enum di `MKV_TRACK_INFO_TYPE_AUDIO` per l'audio).

Se si dispone di dati provati codec per la traccia audio, è possibile passarli quando si chiama la funzione `addTrack`. È inoltre possibile inviare i dati privati del codec dopo aver creato l'oggetto durante la chiamata al metodo `start` in `KinesisVideoStream`.

Inserimento di un frame nel flusso video Kinesis

I file multimediali vengono inseriti nel flusso video di Kinesis

utilizzando `KinesisVideoStream::putFrame`, passando un `Frame` oggetto che contiene l'intestazione e i dati multimediali. L'esempio richiama `putFrame` nel file `ProducerApiTest.cpp`:

```
frame.duration = FRAME_DURATION_IN_MICROS * HUNDREDS_OF_NANOS_IN_A_MICROSECOND;
frame.size = sizeof(frameBuffer_);
frame.frameData = frameBuffer_;
memset(frame.frameData, 0x55, frame.size);

while (!stop_producer_) {
    // Produce frames
    timestamp = std::chrono::duration_cast<std::chrono::nanoseconds>(
        std::chrono::system_clock::now().time_since_epoch()).count() /
    DEFAULT_TIME_UNIT_IN_NANOS;
    frame.index = index++;
    frame.decodingTs = timestamp;
    frame.presentationTs = timestamp;

    // Key frame every 50th
    frame.flags = (frame.index % 50 == 0) ? FRAME_FLAG_KEY_FRAME : FRAME_FLAG_NONE;
    ...

    EXPECT_TRUE(kinesis_video_stream->putFrame(frame));
}
```

Note

L'esempio di producer C++ precedente invia un buffer di dati di prova. In un'applicazione reale, è consigliabile ottenere il buffer e la dimensione del fotogramma dai dati del fotogramma da un'origine multimediale (ad esempio una fotocamera).

L'oggetto `Frame` dispone dei campi seguenti:

- Indice dei fotogrammi. Deve essere un valore a incremento monotono.
- Flag associati al fotogramma. Ad esempio, se il codificatore è stato configurato per ottenere un fotogramma chiave, a questo fotogramma verrà assegnato il contrassegno `FRAME_FLAG_KEY_FRAME`.
- Timestamp di decodifica.

- Timestamp della presentazione.
- Durata del fotogramma (in unità di 100 ns).
- Dimensione del fotogramma in byte.
- Dati del fotogramma.

Per ulteriori informazioni sul formato del frame, consulta [Kinesis Video Streams Data Model](#).

Inserire un KinesisVideoFrame in una traccia specifica di KinesisVideoStream

È possibile utilizzare la `PutFrameHelper` classe per inserire i dati del frame in una traccia specifica. Innanzitutto, chiama il `getFrameData Buffer` per ottenere un puntatore a uno dei buffer preallocati per inserire i dati. `KinesisVideoFrame` Quindi, puoi chiamare `putFrameMulti Track` per inviare il `KinesisVideoFrame` insieme al valore booleano per indicare il tipo di dati del frame. Utilizza "true" se si tratta di dati video o "false" se il frame contiene dati audio. Il metodo `putFrameMulti Track` utilizza un meccanismo di accodamento per garantire che i frammenti MKV mantengano timestamp dei frame crescenti in modo monotono e che due frammenti non si sovrappongano. Ad esempio, il timestamp MKV del primo fotogramma di un frammento deve essere sempre maggiore del timestamp MKV dell'ultimo fotogramma del frammento precedente.

Ha i seguenti campi: `PutFrameHelper`

- Numero massimo di frame audio nella coda.
- Numero massimo di fotogrammi video nella coda.
- Dimensione da allocare per un singolo frame audio.
- Dimensioni da allocare per un singolo fotogramma video.

Metriche e registrazione delle metriche

L'SDK del producer C++ include funzionalità per i parametri e la registrazione dei parametri.

Puoi utilizzare le operazioni `getKinesisVideoMetrics` e `getKinesisVideoStreamMetrics` API per recuperare informazioni su Kinesis Video Streams e sui tuoi stream attivi.

Il seguente codice è tratto dal file `kinesis-video-pic/src/client/include/com/amazonaws/kinesis/video/client/Include.h`.

```
/**  
 * Gets information about the storage availability.
```

```

*
* @param 1 CLIENT_HANDLE - the client object handle.
* @param 2 PKinesisVideoMetrics - OUT - Kinesis Video metrics to be filled.
*
* @return Status of the function call.
*/
PUBLIC_API STATUS getKinesisVideoMetrics(CLIENT_HANDLE, PKinesisVideoMetrics);

/**
* Gets information about the stream content view.
*
* @param 1 STREAM_HANDLE - the stream object handle.
* @param 2 PStreamMetrics - Stream metrics to fill.
*
* @return Status of the function call.
*/
PUBLIC_API STATUS getKinesisVideoStreamMetrics(STREAM_HANDLE, PStreamMetrics);

```

L'oggetto `PClientMetrics` compilato da `getKinesisVideoMetrics` contiene le seguenti informazioni:

- `contentStoreSize`: La dimensione complessiva in byte dell'archivio di contenuti (la memoria utilizzata per archiviare i dati in streaming).
- `contentStoreAvailableDimensione`: la memoria disponibile nell'archivio dei contenuti, in byte.
- `contentStoreAllocatedDimensione`: la memoria allocata nell'archivio dei contenuti.
- `totalContentViewsDimensione`: la memoria totale utilizzata per la visualizzazione dei contenuti. La visualizzazione dei contenuti è una serie di indici di informazioni presenti nell'archivio dei contenuti.
- `totalFrameRate`: il numero aggregato di frame al secondo in tutti gli stream attivi.
- `totalTransferRate`: Il totale di bit al secondo (bps) inviati in tutti i flussi.

L'oggetto `PStreamMetrics` compilato da `getKinesisVideoStreamMetrics` contiene le seguenti informazioni:

- `currentViewDuration`: La differenza in unità di 100 ns tra l'inizio della vista del contenuto (quando i frame sono codificati) e la posizione corrente (quando i dati dei frame vengono inviati a Kinesis Video Streams).
- `overallViewDuration`: La differenza in unità di 100 ns tra l'inizio della vista del contenuto (quando i frame sono codificati) e la parte finale (quando i frame vengono eliminati dalla memoria, o

perché viene superato lo spazio totale allocato per la visualizzazione del contenuto o perché viene `PersistedAck` ricevuto un messaggio da Kinesis Video Streams e i frame noti per essere persistenti vengono svuotati).

- `currentViewSize`: La dimensione in byte della visualizzazione del contenuto dall'intestazione (quando i frame sono codificati) alla posizione corrente (quando i frame vengono inviati a Kinesis Video Streams).
- `overallViewSize`: La dimensione totale in byte della visualizzazione del contenuto.
- `currentFrameRate`: L'ultima velocità misurata dello stream, in fotogrammi al secondo.
- `currentTransferRate`: L'ultima velocità misurata dello stream, in byte al secondo.

Teardown:

Se desideri inviare i byte rimanenti in un buffer e attendere il ACK, puoi utilizzare `stopSync`:

```
kinesis_video_stream->stopSync();
```

In alternativa, puoi chiamare `stop` per terminare lo streaming:

```
kinesis_video_stream->stop();
```

Dopo aver arrestato il flusso, puoi liberarlo richiamando la seguente API:

```
kinesis_video_producer_->freeStream(kinesis_video_stream);
```

Approfondimenti

[the section called “Passaggio 3: Esegui e verifica il codice”](#)

Passaggio 3: Esegui e verifica il codice

Per eseguire e verificare il codice per la [procedura C++ Producer Library](#), consulta le seguenti istruzioni specifiche del sistema operativo:

- [Linux](#)
- [macOS](#)
- [Windows](#)
- [Sistema operativo Raspberry Pi](#)

Puoi monitorare il traffico sul tuo stream guardando le metriche associate al tuo stream nella CloudWatch console Amazon, ad esempio `PutMedia.IncomingBytes`.

Usare C++ Producer SDK come plugin GStreamer

[GStreamer](#) è un popolare framework multimediale utilizzato da più fotocamere e sorgenti video per creare pipeline multimediali personalizzate combinando plugin modulari. Il plugin Kinesis Video Streams GStreamer semplifica l'integrazione della pipeline multimediale GStreamer esistente con Kinesis Video Streams.

Per informazioni sull'uso dell'SDK di C++ Producer come plugin GStreamer, consulta [Esempio: plugin GStreamer SDK di Kinesis Video Streams Producer - kvssink](#).

Utilizzo del C++ Producer SDK come plugin GStreamer in un contenitore Docker

[GStreamer](#) è un popolare framework multimediale utilizzato da più telecamere e sorgenti video per creare pipeline multimediali personalizzate combinando plugin modulari. Il plugin Kinesis Video Streams GStreamer semplifica l'integrazione della pipeline multimediale GStreamer esistente con Kinesis Video Streams.

Inoltre, l'utilizzo di [Docker](#) per creare la pipeline GStreamer standardizza l'ambiente operativo per Kinesis Video Streams, semplificando la creazione e l'esecuzione dell'applicazione.

Per informazioni sull'uso dell'SDK di C++ Producer come plugin GStreamer in un container Docker, consulta [Esegui l'elemento GStreamer in un contenitore Docker](#).

Utilizzo della registrazione con C++ Producer SDK

Puoi configurare il logging per le applicazioni dell'SDK di C++ Producer SDK nel file `kvs_log_configuration` nella cartella `kinesis-video-native-build`.

L'esempio seguente mostra la prima riga del file di configurazione predefinito, che configura l'applicazione per scrivere voci di livello DEBUG nella AWS Management Console:

```
log4cplus.rootLogger=DEBUG, KvsConsoleAppender
```

È possibile impostare il livello di logging su INFO per ottenere un log meno dettagliato.

Per configurare l'applicazione in modo che scriva le voci di registro in un file di registro, aggiorna la prima riga del file nel modo seguente:

```
log4cplus.rootLogger=DEBUG, KvsConsoleAppender, KvsFileAppender
```

Ciò consente di configurare l'applicazione per scrivere voci di log nel file `kvs.log` nella cartella `kinesis-video-native-build/log`.

Per modificare la posizione del file di log, aggiorna la riga seguente con il nuovo percorso:

```
log4cplus.appender.KvsFileAppender.File=../log/kvs.log
```

Note

Se il logging di livello DEBUG viene scritto in un file, il file di log può occupare rapidamente tutto lo spazio di storage disponibile nel dispositivo.

Utilizzo di C Producer Library

Puoi utilizzare la C Producer Library fornita da Amazon Kinesis Video Streams per scrivere codice applicativo per inviare dati multimediali da un dispositivo a un flusso video Kinesis.

Modello a oggetti

[La libreria Kinesis Video Streams C Producer si basa su un componente comune chiamato Platform Independent Codebase \(PIC\), GitHub disponibile all'indirizzo https://github.com/aws-labs/-pic/.](https://github.com/aws-labs/-pic/) Il PIC contiene una logica di business indipendente dalla piattaforma per i componenti fondamentali. La Kinesis Video Streams C Producer Library include PIC con un ulteriore livello di API che consente callback ed eventi specifici per scenari e piattaforme. La libreria Kinesis Video Streams C Producer ha i seguenti componenti basati su PIC:

- Provider di informazioni sui dispositivi: espone la `DeviceInfo` struttura che può essere fornita direttamente all'API PIC. È possibile configurare un set di provider, incluso un provider ottimizzato

per gli scenari applicativi in grado di ottimizzare l'archivio di contenuti in base al numero e ai tipi di flussi gestiti dall'applicazione e alla quantità di buffering richiesta configurata in base alla quantità di RAM disponibile.

- **Stream info provider:** espone la `StreamInfo` struttura che può essere fornita direttamente all'API PIC. Esiste una serie di provider specifici per i tipi di applicazioni e i tipi più comuni di scenari di streaming. Questi includono provider come video, audio e multitraccia audio e video. Ciascuno di questi scenari presenta impostazioni predefinite che è possibile personalizzare in base ai requisiti dell'applicazione.
- **Provider di callback:** espone la `ClientCallbacks` struttura che può essere fornita direttamente all'API PIC. Ciò include un set di provider di callback per il networking (callback API basati su CURL), l'autorizzazione (API delle AWS credenziali) e lo streaming di nuovi tentativi in caso di callback di errori. L'API Callback Provider richiede una serie di argomenti per la configurazione, come le informazioni di autorizzazione e. Regione AWS Questa operazione viene eseguita utilizzando certificati IoT o utilizzando AWS AccessKeyId SecretKey, o SessionToken. È possibile potenziare i provider di callback con callback personalizzati se l'applicazione ha bisogno di ulteriori elaborazioni di un determinato callback per raggiungere la logica specifica dell'applicazione.
- **FrameOrderCoordinator**— Aiuta a gestire la sincronizzazione audio e video per scenari multitraccia. Ha un comportamento predefinito, che è possibile personalizzare per gestire la logica specifica dell'applicazione. Inoltre, semplifica l'imballaggio dei metadati dei frame nella struttura PIC Frame prima di inviarli all'API PIC di livello inferiore. Per gli scenari non multitraccia, questo componente è un punto di passaggio all'API `putFrame` PIC.

La libreria C fornisce i seguenti oggetti per gestire l'invio di dati a un flusso video di Kinesis:

- **KinesisVideoClient**— Contiene informazioni sul dispositivo e mantiene i callback per segnalare gli eventi di Kinesis Video Streams.
- **KinesisVideoStream**— Rappresenta informazioni sui parametri del flusso video, come nome, periodo di conservazione dei dati e tipo di contenuto multimediale.

Inserimento di contenuti multimediali nello stream

È possibile utilizzare i metodi forniti dalla libreria C (ad esempio `PutKinesisVideoFrame`) per inserire dati nell'`KinesisVideoStream` oggetto. La libreria gestisce quindi lo stato interno dei dati, che può includere le seguenti attività:

- Esecuzione dell'autenticazione.

- Controllo della latenza di rete. Se la latenza è troppo elevata, la libreria potrebbe scegliere di eliminare i fotogrammi.
- Monitoraggio dello stato dello streaming in corso.

Procedura: utilizzo dell'SDK del producer C

Questa procedura dimostra come utilizzare il client Kinesis Video Streams e le sorgenti multimediali in un'applicazione C per inviare fotogrammi video con codifica H.264 al flusso video Kinesis.

La procedura include le seguenti fasi:

- [Passaggio 1: scarica il codice di C Producer Library](#)
- [Fase 2: Scrivere ed esaminare il codice](#)
- [Passaggio 3: Esegui e verifica il codice](#)

Prerequisiti

- **Credenziali:** nel codice di esempio, fornisci le credenziali specificando un profilo che hai impostato nel file di profilo delle credenziali. AWS Se non lo hai già fatto, configura prima il profilo delle credenziali.

Per ulteriori informazioni, consulta [Configurare le AWS credenziali](#) e la regione per lo sviluppo.

- **Integrazione dell'archivio certificati:** la Kinesis Video Streams Producer Library deve stabilire un rapporto di fiducia con il servizio che chiama. Ciò avviene tramite la convalida delle autorità di certificazione (CA) nell'archivio di certificati pubblico. Sui modelli basati su Linux, questo store si trova nella directory `/etc/ssl/`.

Scarica il certificato dal percorso seguente per lo store dei certificati:

<https://www.amazontrust.com/repository/SFSRootCAG2.pem>

- Installa le seguenti dipendenze di build per macOS:
 - [Autoconf 2.69](#) (Licenza GPLv3+/Autoconf: GNU GPL versione 3 o successive)
 - [CMake 3.7 o 3.8](#)
 - [Pkg-Config](#)
 - xCode (macOS) / clang / gcc (xcode-select versione 2347)
 - Java Development Kit (JDK) (per la compilazione di Java JNI)

- [Lib-Pkg](#)
- Installa le seguenti dipendenze di build per Ubuntu:
 - Git: `sudo apt install git`
 - [CMake](#): `sudo apt install cmake`
 - G++: `sudo apt install g++`
 - configurazione pkg: `sudo apt install pkg-config`
 - OpenJDK: `sudo apt install openjdk-8-jdk`
 - Imposta la variabile di ambiente JAVA_HOME: `export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/`

Approfondimenti

[Passaggio 1: scarica il codice di C Producer Library](#)

Passaggio 1: scarica il codice di C Producer Library

In questa sezione sarà possibile scaricare le librerie di basso livello. Per i prerequisiti e altri dettagli su questo esempio, consulta [Utilizzo di C Producer Library](#).

1. Crea una directory, quindi clona il codice sorgente di esempio dal GitHub repository.

```
git clone --recursive https://github.com/aws-labs/amazon-kinesis-video-streams-producer-c.git
```

Note

Se non si esegue Git clone con `--recursive`, eseguire `git submodule update --init` nella directory `amazon-kinesis-video-streams-producer-c/open-source`. È inoltre necessario installare `pkg-config`, `CMake` e un ambiente di compilazione.

[Per ulteriori informazioni, vedere https://github.com/aws-labs/amazon-kinesis-video-streams-producer-c.git#readme](https://github.com/aws-labs/amazon-kinesis-video-streams-producer-c.git#readme).

2. Apri il codice nell'ambiente di sviluppo integrato (IDE) di tua scelta (ad esempio, [Eclipse](#)).

Approfondimenti

[Fase 2: Scrivere ed esaminare il codice](#)

Fase 2: Scrivere ed esaminare il codice

In questa sezione, si esamina il codice dell'applicazione di esempio `KvsVideoOnlyStreamingSample.c` nella `samples` cartella del repository <https://github.com/aws-labs/amazon-kinesis-video-streams-producer-c> on GitHub. Hai scaricato questo codice nella fase precedente. Questo esempio dimostra come utilizzare la C Producer Library per inviare frame video con codifica H.264 all'interno della `samples/h264SampleFrames` cartella al flusso video Kinesis.

Questa applicazione di esempio ha tre sezioni:

- Inizializzazione e configurazione:
 - Inizializzazione e configurazione della pipeline dei supporti specifici della piattaforma.
 - Inizializzazione e configurazione `KinesisVideoClient` e `KinesisVideoStream` per la pipeline, impostazione dei callback, integrazione dell'autenticazione specifica dello scenario, estrazione e invio di dati privati del codec e trasferimento dello stream allo stato `READY`.
- Loop principale:
 - Ottenere il frame dalla pipeline dei supporti con timestamp e flag.
 - `KinesisVideoStream` invio del frame a.
- Teardown:
 - Arresto (sincronizzazione) `KinesisVideoStream`, liberazione, liberazione `KinesisVideoStream`.
`KinesisVideoClient`

Questa applicazione di esempio completa le attività seguenti:

- Chiama l'API `createDefaultDeviceInfo` per creare l'oggetto `deviceInfo` che contiene informazioni sul dispositivo o sulla configurazione dello storage.

```
// default storage size is 128MB. Use setDeviceInfoStorageSize after create to change storage size.
CHK_STATUS(createDefaultDeviceInfo(&pDeviceInfo));
// adjust members of pDeviceInfo here if needed
pDeviceInfo->clientInfo.loggerLogLevel = LOG_LEVEL_DEBUG;
```

- Chiama l'API `createRealtimeVideoStreamInfoProvider` per creare l'oggetto `StreamInfo`.

```
CHK_STATUS(createRealtimeVideoStreamInfoProvider(streamName,  
    DEFAULT_RETENTION_PERIOD, DEFAULT_BUFFER_DURATION, &pStreamInfo));  
// adjust members of pStreamInfo here if needed
```

- Chiama l'`createDefaultCallbacksProviderWithAwsCredentialsAPI` per creare il provider di callback predefinito basato su credenziali statiche. AWS

```
CHK_STATUS(createDefaultCallbacksProviderWithAwsCredentials(accessKey,  
    secretKey,  
    sessionToken,  
    MAX_UINT64,  
    region,  
    cacertPath,  
    NULL,  
    NULL,  
    FALSE,  
    &pClientCallbacks));
```

- Chiama l'`createKinesisVideoClientAPI` per creare l'`KinesisVideoClient` oggetto che contiene informazioni sull'archiviazione del dispositivo e mantiene i callback per segnalare gli eventi di Kinesis Video Streams.

```
CHK_STATUS(createKinesisVideoClient(pDeviceInfo, pClientCallbacks, &clientHandle));
```

- Chiama l'API `createKinesisVideoStreamSync` per creare l'oggetto `KinesisVideoStream`.

```
CHK_STATUS(createKinesisVideoStreamSync(clientHandle, pStreamInfo, &streamHandle));
```

- Configura un frame di esempio e chiama l'API `PutKinesisVideoFrame` per l'invio di frame all'oggetto `KinesisVideoStream`.

```
// setup sample frame
MEMSET(frameBuffer, 0x00, frameSize);
frame.frameData = frameBuffer;
frame.version = FRAME_CURRENT_VERSION;
frame.trackId = DEFAULT_VIDEO_TRACK_ID;
frame.duration = HUNDREDS_OF_NANOS_IN_A_SECOND / DEFAULT_FPS_VALUE;
frame.decodingTs = defaultGetTime(); // current time
frame.presentationTs = frame.decodingTs;

while(defaultGetTime() > streamStopTime) {
    frame.index = frameIndex;
    frame.flags = fileIndex % DEFAULT_KEY_FRAME_INTERVAL == 0 ?
FRAME_FLAG_KEY_FRAME : FRAME_FLAG_NONE;
    frame.size = sizeof(frameBuffer);

    CHK_STATUS(readFrameData(&frame, frameFilePath));

    CHK_STATUS(putKinesisVideoFrame(streamHandle, &frame));
    defaultThreadSleep(frame.duration);

    frame.decodingTs += frame.duration;
    frame.presentationTs = frame.decodingTs;
    frameIndex++;
    fileIndex++;
    fileIndex = fileIndex % NUMBER_OF_FRAME_FILES;
}
```

- Teardown:

```
CHK_STATUS(stopKinesisVideoStreamSync(streamHandle));
CHK_STATUS(freeKinesisVideoStream(&streamHandle));
CHK_STATUS(freeKinesisVideoClient(&clientHandle));
```

Approfondimenti

[Passaggio 3: Esegui e verifica il codice](#)

Passaggio 3: Esegui e verifica il codice

Per eseguire e verificare il codice per la [procedura per C Producer Library](#), esegui le operazioni descritte di seguito:

1. Esegui i seguenti comandi per creare una build directory nell'[SDK C scaricato](#) e avviarla cmake da essa:

```
mkdir -p amazon-kinesis-video-streams-producer-c/build;  
cd amazon-kinesis-video-streams-producer-c/build;  
cmake ..
```

È possibile inviare le seguenti opzioni a `cmake ..`

- `-DBUILD_DEPENDENCIES`- se creare librerie dipendenti dai sorgenti.
- `-DBUILD_TEST=TRUE`- build unit e test di integrazione. Potrebbe essere utile per confermare il supporto per il tuo dispositivo.

```
./tst/webrtc_client_test
```

- `-DCODE_COVERAGE`- abilita la segnalazione della copertura.
 - `-DCOMPILER_WARNINGS`- abilita tutti gli avvisi del compilatore.
 - `-DADDRESS_SANITIZER`- costruisci con AddressSanitizer
 - `-DMEMORY_SANITIZER`- costruisci con MemorySanitizer.
 - `-DTHREAD_SANITIZER`- costruisci con ThreadSanitizer.
 - `-DUNDEFINED_BEHAVIOR_SANITIZER`- costruisci con UndefinedBehaviorSanitizer.
 - `-DALIGNED_MEMORY_MODEL`: compilazione per dispositivi solo con modello di memoria allineato Il valore predefinito è OFF.
2. Passa alla build directory che hai appena creato con il passaggio precedente ed esegui `make` per creare l'SDK WebRTC C e gli esempi forniti.

```
make
```

3. L'applicazione di esempio `kinesis_video_cproducer_video_only_sample` invia fotogrammi video con codifica h.264 all'interno della cartella a Kinesis `samples/h264SampleFrames` Video Streams. Il seguente comando invia i fotogrammi video in loop per dieci secondi a Kinesis Video Streams:

```
./kinesis_video_cproducer_video_only_sample YourStreamName 10
```

Se desideri inviare frame con codifica H.264 da un'altra cartella (ad esempio, `MyH264FramesFolder`), esegui l'esempio con i seguenti argomenti:

```
./kinesis_video_cproducer_video_only_sample YourStreamName 10 MyH264FramesFolder
```

4. Per abilitare i log verbose, crea le definizioni C `HEAP_DEBUG` e `LOG_STREAMING` rimuovendo i commenti dalla righe appropriate in `CMakeList.txt`.

Puoi monitorare l'avanzamento della suite di test nell'output di debug nell'IDE. Puoi anche monitorare il traffico sul tuo stream guardando le metriche associate al tuo stream nella CloudWatch console Amazon, ad esempio `PutMedia.IncomingBytes`.

Note

Poiché il test harness invia solo fotogrammi di byte vuoti, la console non visualizza i dati come un flusso di video.

Uso dell'SDK del producer C++ su Raspberry Pi

Raspberry Pi è un piccolo computer economico che può essere utilizzato per insegnare e apprendere competenze di programmazione di base. Questo tutorial descrive come configurare e utilizzare l'SDK Amazon Kinesis Video Streams C++ Producer su un dispositivo Raspberry Pi. Le fasi includono anche come verificare l'installazione utilizzando l'applicazione demo `GStreamer`.

Argomenti

- [Prerequisiti](#)

- [Crea un utente IAM con il permesso di scrivere su Kinesis Video Streams](#)
- [Collega il tuo Raspberry Pi alla tua rete Wi-Fi](#)
- [Connect da remoto al tuo Raspberry Pi](#)
- [Configura la fotocamera Raspberry Pi](#)
- [Installa i prerequisiti del software](#)
- [Scarica e crea l'SDK Kinesis Video Streams C++ Producer](#)
- [Trasmetti video al tuo stream video Kinesis e guarda il live streaming](#)

Prerequisiti

Prima di configurare l'SDK del producer C++ su Raspberry Pi, accertati di soddisfare i seguenti prerequisiti:

- Un dispositivo Raspberry Pi con la seguente configurazione:
 - Versione della scheda: 3 Model B o versioni successive.
 - Un modulo fotocamera connesso.
 - Una scheda SD con una capacità minima di 8 GB.
 - Sistema operativo Raspbian (kernel 4.9 o versioni successive) installato. [Puoi scaricare l'immagine più recente del sistema operativo Raspberry Pi \(precedentemente chiamato Raspbian\) dal sito Web di Raspberry Pi](#). Segui le istruzioni di Raspberry Pi per [installare l'immagine scaricata su una scheda SD](#).
- E Account AWS con un flusso video Kinesis. Per ulteriori informazioni, consulta [Nozioni di base su Kinesis Video Streams](#).

Note

L'SDK C++ Producer utilizza la regione Stati Uniti occidentali (Oregon) (*us-west-2*) per impostazione predefinita. Per utilizzare l'impostazione predefinita, Regione AWS crea il tuo streaming video Kinesis nella regione Stati Uniti occidentali (Oregon).

Per utilizzare una regione diversa per lo streaming video di Kinesis, esegui una delle seguenti operazioni:

- Impostare le seguenti variabili di ambiente nella regione (per esempio, *us-east-1*):

```
export AWS_DEFAULT_REGION=us-east-1
```

Crea un utente IAM con il permesso di scrivere su Kinesis Video Streams

Se non l'hai già fatto, configura un utente AWS Identity and Access Management (IAM) con le autorizzazioni per scrivere su un flusso video Kinesis.

Queste procedure hanno lo scopo di aiutarti a iniziare rapidamente a utilizzare una coppia di chiavi di AWS accesso. I dispositivi possono utilizzare certificati X.509 a cui connettersi. AWS IoT [the section called "Controllo dell'accesso alle risorse Kinesis Video Streams tramite AWS IoT"](#) Per ulteriori informazioni su come configurare il dispositivo per l'utilizzo dell'autenticazione basata su certificati, consulta.

1. [Accedi AWS Management Console e apri la console IAM all'indirizzo https://console.aws.amazon.com/iam/.](https://console.aws.amazon.com/iam/)
2. Nel menu di navigazione a sinistra, scegli Users (Utenti).
3. Per creare un nuovo utente, scegli Add user (Aggiungi utente).
4. Fornire un User name (Nome utente) descrittivo per l'utente, ad esempio **kinesis-video-raspberry-pi-producer**.
5. Sotto Access type (Tipo di accesso), scegli Programmatic access (Accesso programmatico).
6. Scegli Successivo: autorizzazioni.
7. In Imposta le autorizzazioni per kinesis-video-raspberry-pi-producer, scegli Allega direttamente le politiche esistenti.
8. Scegli Crea policy. La pagina Create policy (Crea policy) viene visualizzata in una nuova scheda del browser Web.
9. Scegli la scheda JSON.
10. Copia la seguente policy JSON e incollala nell'area del testo. Questa politica concede all'utente l'autorizzazione a creare e scrivere dati nei flussi video Kinesis.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "kinesisvideo:DescribeStream",
      "kinesisvideo:CreateStream",
      "kinesisvideo:GetDataEndpoint",
      "kinesisvideo:PutMedia"
    ]
  }]
}
```

```
    ],
    "Resource": [
      "*"
    ]
  }]
}
```

11. Scegli Verifica policy.
12. Fornire un Name (Nome) alla policy, ad esempio **kinesis-video-stream-write-policy**.
13. Scegli Crea policy.
14. Torna alla scheda Add user (Aggiungi utente) nel browser e scegli Refresh (Aggiorna).
15. Nella casella di ricerca digita il nome della policy creata.
16. Seleziona la casella di controllo accanto alla nuova policy nell'elenco.
17. Scegliere Next:Review (Successivo:Rivedi).
18. Selezionare Create user (Crea utente).
19. Nella console viene visualizzato l'ID chiave di accesso per il nuovo utente. Scegli Show (Mostra) per visualizzare la Secret Access Key (Chiave di accesso segreta). Registra questi valori perché saranno necessari per la configurazione dell'applicazione.

Collega il tuo Raspberry Pi alla tua rete Wi-Fi

Puoi utilizzare il Raspberry Pi in modalità headless, ossia senza tastiera, monitor o cavo di rete collegato. Se usi un monitor e una tastiera collegati, passa a [Configura la fotocamera Raspberry Pi](#).

1. Sul tuo computer, crea un file denominato `wpa_supplicant.conf`.
2. Copia il seguente testo e incollalo nel `wpa_supplicant.conf` file:

```
country=US
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1

network={
  ssid="Your Wi-Fi SSID"
  scan_ssid=1
  key_mgmt=WPA-PSK
  psk="Your Wi-Fi Password"
}
```

Sostituisci i valori `ssid` e `psk` con le informazioni per la rete Wi-Fi.

3. Copia il file `wpa_supplicant.conf` nella scheda SD. Devi copiarlo nella directory principale del volume boot.
4. Inserisci la scheda SD nel Raspberry Pi e accendi il dispositivo. Il dispositivo viene aggiunto alla rete Wi-Fi e SSH viene abilitato.

Connect da remoto al tuo Raspberry Pi

Puoi connetterti in remoto al Raspberry Pi in modalità headless. Se utilizzi il Raspberry Pi con un monitor e una tastiera collegati, passa a [Configura la fotocamera Raspberry Pi](#).

1. Prima di connetterti al dispositivo Raspberry Pi in remoto, esegui una delle operazioni seguenti per determinarne l'indirizzo IP:
 - Se disponi dell'accesso al router Wi-Fi della rete, controlla i dispositivi Wi-Fi connessi. Trova il dispositivo denominato `Raspberry Pi` per trovare l'indirizzo IP del dispositivo.
 - Se non disponi dell'accesso al router Wi-Fi della rete, puoi usare altri prodotti software per trovare i dispositivi in rete. [Fing](#) è un'applicazione molto diffusa disponibile per dispositivi sia Android che iOS. Puoi utilizzare la versione gratuita di questa applicazione per trovare gli indirizzi IP dei dispositivi in rete.
2. Una volta trovato l'indirizzo IP del dispositivo Raspberry Pi, puoi usare qualsiasi applicazione terminale per connetterti.
 - Su macOS o Linux, usa `ssh`:

```
ssh pi@<IP address>
```

 - Su Windows, usa [PuTTY](#), un client SSH gratuito per Windows.

Per una nuova installazione di Raspbian, il nome utente è **pi** e la password è **raspberry**. Ti consigliamo di [modificare la password predefinita](#).

Configura la fotocamera Raspberry Pi

Segui questi passaggi per configurare la videocamera Raspberry Pi per inviare video dal dispositivo a un flusso video Kinesis.

1. Apri un editor per aggiornare il file `modules` con il comando seguente:

```
sudo nano /etc/modules
```

2. Aggiungi la riga seguente alla fine del file, se non è già presente:

```
bcm2835-v4l2
```

3. Salva il file ed esci dall'editor di testo (Ctrl-X).
4. Riavvia il dispositivo Raspberry Pi:

```
sudo reboot
```

5. Quando il dispositivo si riavvia, esegui di nuovo la connessione utilizzando l'applicazione terminale se ti connetti in remoto.
6. Aprire `raspi-config`:

```
sudo raspi-config
```

7. Scegli Opzioni di interfaccia, Legacy Camera. Nelle versioni precedenti del sistema operativo Raspbian, questa opzione di menu potrebbe trovarsi in Opzioni di interfaccia, Fotocamera.

Abilita la fotocamera se non è già abilitata, quindi riavvia se viene richiesto.

8. Verifica che la fotocamera funzioni digitando il comando seguente:

```
raspistill -v -o test.jpg
```

Se la fotocamera è configurata correttamente, questo comando acquisisce un'immagine dalla fotocamera, la salva in un file denominato e visualizza messaggi `test.jpg` informativi.

Installa i prerequisiti del software

L'SDK del producer C++ richiede l'installazione dei seguenti prerequisiti software su Raspberry Pi.

1. Aggiorna l'elenco dei pacchetti e installa le librerie necessarie per creare l'SDK. Digita i seguenti comandi:

```
sudo apt update
sudo apt install -y \
```

```
automake \  
build-essential \  
cmake \  
git \  
gststreamer1.0-plugins-base-apps \  
gststreamer1.0-plugins-bad \  
gststreamer1.0-plugins-good \  
gststreamer1.0-plugins-ugly \  
gststreamer1.0-tools \  
gststreamer1.0-omx-generic \  
libcurl4-openssl-dev \  
libgststreamer1.0-dev \  
libgststreamer-plugins-base1.0-dev \  
liblog4cplusplus-dev \  
libssl-dev \  
pkg-config
```

2. Copia il file PEM seguente in `/etc/ssl/cert.pem`:

```
sudo curl https://www.amazontrust.com/repository/AmazonRootCA1.pem -o /etc/ssl/  
AmazonRootCA1.pem  
sudo chmod 644 /etc/ssl/AmazonRootCA1.pem
```

Scarica e crea l'SDK Kinesis Video Streams C++ Producer

È possibile scaricare e creare l'SDK di Kinesis Video Streams C++ Producer utilizzando la seguente procedura. Questo approccio richiede più tempo per la compilazione, a seconda della connettività di rete e della velocità del processore.

1. Scarica l'SDK. Type:

```
git clone https://github.com/aws-labs/amazon-kinesis-video-streams-producer-sdk-  
cpp.git
```

2. Prepara una directory di compilazione. Type:

```
mkdir -p amazon-kinesis-video-streams-producer-sdk-cpp/build  
cd amazon-kinesis-video-streams-producer-sdk-cpp/build
```

3. Crea l'SDK e le applicazioni di esempio. A seconda del modello di Raspberry Pi su cui stai costruendo, l'esecuzione per la prima volta potrebbe richiedere diverse ore:

```
cmake .. -DBUILD_GSTREAMER_PLUGIN=ON -DBUILD_DEPENDENCIES=FALSE
make
```

Trasmetti video al tuo stream video Kinesis e guarda il live streaming

1. Per eseguire l'applicazione di esempio sono necessarie le seguenti informazioni:
 - Il nome del flusso creato nella sezione [Prerequisiti](#).
 - Le credenziali dell'account (ID chiave di accesso e chiave di accesso segreta) create in [Crea un utente IAM con il permesso di scrivere su Kinesis Video Streams](#).
2. Esegui l'applicazione di esempio utilizzando i seguenti comandi. Sostituisci i segnaposto con valori per il tuo ambiente.

```
export GST_PLUGIN_PATH=Directory Where You Cloned the SDK/amazon-kinesis-video-streams-producer-sdk-cpp/build
export AWS_DEFAULT_REGION=AWS Region i.e. us-east-1
export AWS_ACCESS_KEY_ID=Access Key ID
export AWS_SECRET_ACCESS_KEY=Secret Access Key
./kvs_gstreamer_sample Your Stream Name
```

3. Se l'applicazione di esempio si chiude con un `library not found` errore, digitate il comando seguente per verificare che il progetto sia collegato correttamente alle sue dipendenze open source:

```
gst-inspect-1.0 kvssink
```

4. Apri la console [Kinesis Video Streams](#).
5. Scegli il nome per il flusso creato.

Il flusso di video inviato da Raspberry Pi viene visualizzato nella console.

Durante la riproduzione dello streaming, puoi sperimentare le seguenti funzionalità della console Kinesis Video Streams:

- Nella sezione Video preview (Anteprima video) usa i controlli di navigazione per riavvolgere o far avanzare rapidamente il flusso.

- Nella sezione Stream info (Informazioni flusso) annotare il codec, la risoluzione e la velocità in bit del flusso. I valori della risoluzione e della velocità in bit sono volutamente bassi su Raspberry Pi per ridurre al minimo l'utilizzo della larghezza di banda per questo tutorial. Per visualizzare le CloudWatch metriche Amazon che vengono create per il tuo stream, scegli Visualizza le metriche dello stream in. CloudWatch
- Sotto Data retention period (Periodo di retention dei dati), nota che il flusso di video viene conservato per un giorno. Puoi modificare questo valore e impostarlo su No data retention (Nessuna retention dei dati) oppure impostare un valore compreso tra un giorno e diversi anni.

Con la crittografia lato server, tieni presente che i tuoi dati vengono crittografati quando sono inattivi utilizzando una chiave gestita da (). AWS Key Management Service AWS KMS

Documentazione di riferimento per l'SDK producer

Questa sezione contiene informazioni su limiti, codici di errore e altri riferimenti relativi alle [Librerie per produttori di Kinesis Video Streams](#).

Argomenti

- [Limiti di Producer SDK](#)
- [Riferimento al codice di errore](#)
- [Documentazione di riferimento per i flag di adattamento del Network Abstraction Layer \(NAL\)](#)
- [Strutture SDK di Producer](#)
- [Strutture dei flussi video Kinesis](#)
- [Richiamate SDK di Producer](#)

Limiti di Producer SDK

La tabella riportata di seguito contiene i limiti attuali dei valori delle [Producer Library](#).

Note

Prima di impostare questi valori, è necessario convalidare gli input. L'SDK non convalida questi limiti e si verifica un errore di runtime se vengono superati i limiti.

Valore	Limite	Note
Numero massimo di flussi	128	Il numero massimo di flussi che un oggetto producer può creare. Si tratta di un limite flessibile, per il quale puoi richiedere un aumento. Garantisce che il produttore e non crei accidentalmente stream in modo ricorsivo.
Lunghezza massima del nome del dispositivo	128 caratteri	
Numero massimo di tag	50 per flusso	
Lunghezza massima del nome del flusso	256 caratteri	
Dimensione minima di storage	10 MiB = 10 * 1024 * 1024 byte	
Dimensione massima di storage	10 GiB = 10 * 1024 * 1024 * 1024 byte	
Lunghezza massima del percorso della directory root	4,096 caratteri	
Lunghezza massima delle informazioni di autorizzazione	10,000 byte	
Lunghezza massima della stringa URI	10,000 caratteri	
Lunghezza massima del nome del tag	128 caratteri	
Lunghezza massima del valore del tag	1,024 caratteri	

Valore	Limite	Note
Periodo minimo del token di sicurezza	30 secondi	
Periodo di tolleranza del token di sicurezza	40 minuti	Se la durata specificata è più lunga, è limitata a questo valore.
Periodo di conservazione	0 o superiore a un'ora	Il valore 0 indica nessuna conservazione.
Durata minima del cluster	1 secondo	Il valore è specificato in unità di 100 ns, ovvero lo standard dell'SDK.
Durata massima del cluster	30 secondi	Il valore è specificato in unità di 100 ns, ovvero lo standard dell'SDK. L'API di backend può imporre una durata del cluster più breve.
Dimensione massima del frammento	50 MB	Per ulteriori informazioni, consulta Quote di servizio Kinesis Video Streams .
Durata massima del frammento	20 secondi	Per ulteriori informazioni, consulta Quote di servizio Kinesis Video Streams .
Durata massima della connessione	45 minuti	Il back-end chiude la connessione dopo questo intervallo di tempo. L'SDK ruota il token e stabilisce una nuova connessione entro questo intervallo di tempo.

Valore	Limite	Note
Lunghezza massima del segmento ACK	1,024 caratteri	Lunghezza massima del segmento di conferma inviato alla funzione del parser di ACK.
Lunghezza massima della stringa del tipo di contenuto	128 caratteri	
Lunghezza massima della stringa dell'ID codec	32 caratteri	
Lunghezza massima della stringa del nome della traccia	32 caratteri	
Lunghezza massima dei dati privati del codec	1 MiB = 1 * 1024 * 1024 byte	
Lunghezza minima del valore di scala del timecode	100 ns	Il valore di scala minimo del timecode per rappresentare i timestamp del fotogramma a nel cluster MKV risultante. Il valore è specificato in incrementi di 100 ns, ovvero lo standard dell'SDK.
Lunghezza massima del valore di scala del timecode	1 secondo	Il valore di scala massimo del timecode per rappresentare i timestamp del fotogramma a nel cluster MKV risultante. Il valore è specificato in incrementi di 100 ns, ovvero lo standard dell'SDK.
Numero minimo degli elementi di visualizzazione dei contenuti	10	

Valore	Limite	Note
Durata minima del buffer	20 secondi	Il valore è specificato in incrementi di 100 ns, ovvero lo standard dell'SDK.
Lunghezza massima della versione di aggiornamento	128 caratteri	
Lunghezza massima dell'ARN	1024 caratteri	
Lunghezza massima della sequenza dei frammenti	128 caratteri	
Periodo massimo di conservazione	10 anni	

Riferimento al codice di errore

Questa sezione contiene informazioni sui codici di errore e stato delle [Producer Library](#).

Per informazioni sulle soluzioni ai problemi comuni, consulta [Risoluzione dei problemi relativi a Kinesis Video Streams](#).

Argomenti

- [Errori e codici di stato restituiti da PutFrame Callbacks - Platform Independent Code \(PIC\)](#)
- [Errori e codici di stato restituiti dai PutFrame callback - C producer library](#)

Errori e codici di stato restituiti da PutFrame Callbacks - Platform Independent Code (PIC)

Le seguenti sezioni contengono informazioni sugli errori e sullo stato restituite dai callback per l'PutFrameoperazione all'interno del Platform Independent Code (PIC).

Argomenti

- [Codici di errore e di stato restituiti dalla libreria client](#)
- [Codici di errore e di stato restituiti dalla libreria di durata](#)

- [Codici di errore e di stato restituiti dalla libreria comune](#)
- [Codici di errore e di stato restituiti dalla libreria heap](#)
- [Codici di errore e di stato restituiti dalla libreria MKVGen](#)
- [Codici di errore e di stato restituiti dalla libreria Trace](#)
- [Codici di errore e di stato restituiti dalla libreria Utils](#)
- [Codici di errore e di stato restituiti dalla libreria View](#)

Codici di errore e di stato restituiti dalla libreria client

La tabella seguente contiene informazioni sugli errori e sullo stato restituite dai metodi della libreria Kinesis Client Video Streams.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000001	STATUS_MAX_STREAM_COUNT	È stato raggiunto il numero massimo di flussi.	Specificare un numero massimo di flussi più elevato in DeviceInfo come specificato in Limiti di Producer SDK .
0x52000002	STATUS_MIN_STREAM_COUNT	Errore numero minimo di flussi.	Specificate il numero massimo di stream superiore a zero. DeviceInfo
0x52000003	STATUS_INVALID_DEVICE_NAME_LENGTH	Lunghezza del nome del dispositivo non valida.	Fai riferimento alla lunghezza massima del nome del dispositivo in caratteri specificata in Limiti di Producer SDK .
0x52000004	STATUS_INVALID_DEVICE_INFO_VERSION	Versione della struttura DeviceInfo non valida.	Specificare la versione corrente corretta della struttura.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000005	STATUS_MAX_TAG_COUNT	È stato raggiunto il numero massimo di tag.	Fate riferimento al numero massimo di tag corrente specificato in Limiti di Producer SDK .
0x52000006	STATUS_DEVICE_FINGERPRINT_LENGTH		
0x52000007	STATUS_INVALID_CALLBACKS_VERSION	Versione della struttura Callbacks non valida.	Specificare la versione corrente corretta della struttura.
0x52000008	STATUS_INVALID_STREAM_INFO_VERSION	Versione della struttura StreamInfo non valida.	Specificare la versione corrente corretta della struttura.
0x52000009	STATUS_INVALID_STREAM_NAME_LENGTH	Lunghezza del nome del flusso non valida.	Fai riferimento alla lunghezza massima del nome dello stream in caratteri specificata in Limiti di Producer SDK .
0x5200000a	STATUS_INVALID_STORAGE_SIZE	È stata specificata una dimensione di storage non valida.	La dimensione di storage in byte non deve superare i limiti specificati in Limiti di Producer SDK .
0x5200000b	STATUS_INVALID_DIRECTORY_LENGTH	Lunghezza della stringa della directory root non valida.	Fate riferimento alla lunghezza massima del percorso della directory principale specificata in Limiti di Producer SDK .

Codice	Messaggio	Descrizione	Azione consigliata
0x5200000c	STATUS_INVALID_SPLLL_RATIO	Rapporto di spill non valido.	Esprimete il rapporto di fuoriuscita in percentuale da 0 a 100.
0x5200000d	STATUS_INVALID_STORAGE_INFO_VERSION	Versione della struttura StorageInfo non valida.	Specificare la versione corrente corretta della struttura.
0x5200000e	STATUS_INVALID_STREAM_STATE	Lo stato attuale del flusso non consente l'operazione corrente.	In genere, questo errore si verifica quando l'SDK non riesce a raggiungere lo stato richiesto per eseguire l'operazione richiesta. Ad esempio, si verifica se la chiamata API <code>GetStreamingEndpoint</code> non riesce e se l'applicazione client la ignora e continua a inserire i fotogrammi nel flusso.
0x5200000f	STATUS_SERVICE_CALLBACKS_MISSING	Nella struttura <code>Callbacks</code> non sono presenti dei punti di accesso della funzione di alcune funzioni obbligatorie.	Verificate che i callback obbligatori siano implementati nell'applicazione client. Questo errore è esposto solo ai client Platform Independent Code (PIC). C++ e altri wrapper di livello più elevato sono in grado di soddisfare queste chiamate.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000010	STATUS_SERVICE_CALL_NOT_AUTHORIZED_ERROR	Non autorizzato.	Verifica il token di sicurezza, il certificato, l'integrazione del token di sicurezza e la scadenza. Verifica che al token siano associati i diritti corretti. Per le applicazioni di esempio Kinesis Video Streams, verifica che la variabile di ambiente sia impostata correttamente.
0x52000011	STATUS_DESCRIBE_STREAM_CALL_FAILED	Errore dell'API DescribeStream .	Questo errore viene restituito dopo che il nuovo tentativo dell'API DescribeStream ha restituito un errore. Il client PIC restituisce questo errore dopo aver smesso di riprovare.
0x52000012	STATUS_INVALID_DESCRIBE_STREAM_RESPONSE	Struttura DescribeStreamResponse non valida.	La struttura inviata al DescribeStreamResultEvent è nulla o contiene elementi non validi, come un Amazon Resource Name (ARN) nullo.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000013	STATUS_STREAM_IS_BEING_DELETED_ERROR	Il flusso è in fase di eliminazione.	Il flusso in fase di eliminazione ha causato un errore dell'API. Verifica che nessun altro processo stia tentando di eliminare lo stream mentre lo stream è in uso.
0x52000014	STATUS_SERVICE_CALL_INVALID_ARG_ERROR	Sono stati specificati degli argomenti non validi per la chiamata di servizio.	Il backend restituisce questo errore quando un argomento della chiamata di servizio non è valido o quando l'SDK rileva un errore che non può interpretare.
0x52000015	STATUS_SERVICE_CALL_DEVICE_NOT_FOUND_ERROR	Il dispositivo non è stato trovato.	Verifica che il dispositivo non venga eliminato mentre è in uso.
0x52000016	STATUS_SERVICE_CALL_DEVICE_NOT_PROVISIONED_ERROR	Non è stato effettuato il provisioning del dispositivo.	Verifica che il dispositivo sia stato fornito.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000017	STATUS_SERVICE_CALL_RESOURCE_NOT_FOUND_ERROR	Risorsa generica non trovata restituita dal servizio.	Questo errore si verifica quando il servizio non è in grado di individuare la risorsa (ad esempio, un flusso). Tale errore potrebbe avere più significati in contesti diversi, ma la sua causa più probabile è l'utilizzo delle API prima della creazione del flusso. L'utilizzo dell'SDK conferma che lo stream viene creato per primo.
0x52000018	STATUS_INVALID_AUTH_LEN	Lunghezza delle informazioni di autorizzazione non valida.	Fare riferimento ai valori correnti specificati in Limiti di Producer SDK .
0x52000019	STATUS_CREATE_STREAM_CALL_FAILED	La chiamata dell'API <code>CreateStream</code> non è riuscita.	Fare riferimento alla stringa di errore per ulteriori informazioni dettagliate sul motivo della mancata riuscita dell'operazione.
0x5200002a	STATUS_GET_STREAMING_TOKEN_CALL_FAILED	La chiamata <code>GetStreamingToken</code> non è riuscita.	Fare riferimento alla stringa di errore per ulteriori informazioni dettagliate sul motivo della mancata riuscita dell'operazione.

Codice	Messaggio	Descrizione	Azione consigliata
0x5200002b	STATUS_GET_STREAMING_ENDPOINT_CALL_FAILED	La chiamata dell'API <code>GetStreamingEndpoint</code> non è riuscita.	Fare riferimento alla stringa di errore per ulteriori informazioni dettagliate sul motivo della mancata riuscita dell'operazione.
0x5200002c	STATUS_INVALID_URI_LEN	L'API <code>GetStreamingEndpoint</code> ha restituito una lunghezza della stringa URI non valida.	Fare riferimento ai valori massimi correnti specificati in Limiti di Producer SDK .
0x5200002d	STATUS_PUT_STREAM_CALL_FAILED	La chiamata dell'API <code>PutMedia</code> non è riuscita.	Fare riferimento alla stringa di errore per ulteriori informazioni dettagliate sul motivo della mancata riuscita dell'operazione.

Codice	Messaggio	Descrizione	Azione consigliata
0x5200002e	STATUS_STORE_OUT_OF_MEMORY	La memoria dello store dei contenuti è esaurita.	Lo store dei contenuti viene condiviso tra i flussi e deve disporre di capacità sufficiente per l'archiviazione della durata massima di tutti i flussi più di un altro 20% circa di spazio (per la deframmentazione). È importante non superare la capacità di storage. Scegliere i valori di durata massima per flusso corrispondenti alla dimensione di storage cumulativa e alle tolleranze di latenza. Consigliamo di eliminare i frame non appena escono dalla finestra di visualizzazione dei contenuti anziché semplicemente inserirli (pressione della memoria di archiviazione dei contenuti). Questo perché l'eliminazione dei frame avvia i richiami di notifica della pressione del flusso. L'applicazione è in grado di regolare i componenti multimediali di upstream (come il codificatore) per comprimere il bitrate, eliminare i fotogrammi

Codice	Messaggio	Descrizione	Azione consigliata
			i o adottare le misure necessarie.
0x5200002f	STATUS_NO_MORE_DATA_AVAILABLE	Al momento non sono disponibili altri dati per un flusso.	Si tratta di un risultato potenziale valido quando la pipeline multimediale produce i fotogrammi da inviare al servizio più lentamente rispetto a quanto il thread di networking ne consuma. I client di livello superiore (ad esempio, C++, Java o Android) non visualizzano questo avviso perché è gestito internamente.
0x52000030	STATUS_INVALID_TAG_VERSION	Versione della struttura Tag non valida.	Specificare la versione corrente corretta della struttura.
0x52000031	STATUS_SERVICE_UNKNOWN_ERROR	È stato restituito un errore sconosciuto o generico dallo stack di rete.	Consultare i log per ulteriori informazioni dettagliate.
0x52000032	STATUS_SERVICE_RESOURCE_IN_USE_ERROR	Risorsa in uso.	Restituito dal servizio. Per ulteriori informazioni, consulta la Guida di riferimento all'API Kinesis Video Streams.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000033	STATUS_SERVICE_CLIENT_LIMIT_ERROR	Limite del client.	Restituito dal servizio. Per ulteriori informazioni, consulta la Guida di riferimento all'API Kinesis Video Streams.
0x52000034	STATUS_SERVICE_DEVICE_LIMIT_ERROR	Limite del dispositivo.	Restituito dal servizio. Per ulteriori informazioni, consulta la Guida di riferimento all'API Kinesis Video Streams.
0x52000035	STATUS_SERVICE_STREAM_LIMIT_ERROR	Limite del flusso.	Restituito dal servizio. Per ulteriori informazioni, consulta la Guida di riferimento all'API Kinesis Video Streams.
0x52000036	STATUS_SERVICE_RESOURCE_DELETED_ERROR	La risorsa è stata eliminata o è in fase di eliminazione.	Restituito dal servizio. Per ulteriori informazioni, consulta la Guida di riferimento all'API Kinesis Video Streams.
0x52000037	STATUS_SERVICE_TIMEOUT_ERROR	La chiamata di servizio è scaduta.	La chiamata a una determinata API del servizio è risultata in un timeout. Verifica di disporre di una connessione di rete valida. Il PIC ritenterà l'operazione automaticamente.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000038	STATUS_STREAM_READ_CALLBACK_FAILED	Notifica di flusso pronto all'uso.	Questa notifica viene inviata dal PIC al client per indicare che il flusso asincrono è stato creato.
0x52000039	STATUS_DEVICE_TAGS_COUNT_NON_ZERO_TAGS_NULL	Sono stati specificati dei tag non validi.	Il numero di tag non è zero, ma i tag sono vuoti. Verifica che i tag siano specificati o che il conteggio sia zero.
0x5200003a	STATUS_INVALID_STREAM_DESCRIPTION_VERSION	Versione della struttura StreamDescription non valida.	Specificare la versione corrente corretta della struttura.
0x5200003b	STATUS_INVALID_TAG_NAME_LEN	Lunghezza del nome del tag non valida.	Fare riferimento ai limiti relativi al nome del tag specificati in Limiti di Producer SDK .
0x5200003c	STATUS_INVALID_TAG_VALUE_LEN	Lunghezza del valore del tag non valida.	Fare riferimento ai limiti relativi al valore del tag specificati in Limiti di Producer SDK .
0x5200003d	STATUS_TAG_STREAM_CALL_FAILED	Errore dell'API TagResource .	La chiamata dell'API TagResource non è riuscita. Controllare che la connessione di rete sia attiva. Consultare i log per ulteriori informazioni sull'errore.

Codice	Messaggio	Descrizione	Azione consigliata
0x5200003e	STATUS_INVALID_CUSTOM_DATA	Dati personalizzati non validi durante la chiamata alle API del PIC.	Sono stati specificati dei dati personalizzati non validi in una chiamata alle API del PIC. Questo errore può verificarsi soltanto per i client che utilizzano direttamente il PIC.
0x5200003f	STATUS_INVALID_CREATE_STREAM_RESPONSE	Struttura CreateStreamResponse non valida.	La struttura o i relativi campi dei membri non sono validi (ovvero, l'ARN è nullo o di dimensioni superiori rispetto a quelle specificate in Limiti di Producer SDK).
0x52000040	STATUS_CLIENT_AUTH_CALL_FAILED	Autenticazione del client non riuscita.	Il PIC non è riuscito a ottenere le informazioni di autenticazione corrette (AccessKeyId o SecretAccessKey) dopo diversi tentativi. Verificare l'integrazione dell'autenticazione. Le applicazioni di esempio utilizzano le variabili di ambiente per inviare le informazioni sulle credenziali alla libreria del producer C++.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000041	STATUS_GET_CLIENT_TOKEN_CALL_FAILED	Errore di ricezione della chiamata al token di sicurezza.	Questa situazione può verificarsi per i client che utilizzano direttamente il PIC. Dopo un determinato numero di tentativi, la chiamata non riesce con questo errore.
0x52000042	STATUS_CLIENT_PROVISIONING_CALL_FAILED	Errore di provisioning.	Il provisioning non è implementato.
0x52000043	STATUS_CREATE_CLIENT_CALL_FAILED	Impossibile creare il client del producer.	Un errore generico restituito dal PIC dopo un determinato numero di tentativi quando la creazione del client non riesce.
0x52000044	STATUS_CLIENT_READY_CALLBACK_FAILED	Non è stato possibile impostare il client del producer sullo stato READY.	Restituito dal computer di stato PIC se il PIC non riesce a passare allo stato READY. Consultare i log per ulteriori informazioni sulla causa principale.
0x52000045	STATUS_TAG_CLIENT_CALL_FAILED	Errore di TagResource per il client del producer.	Errore della chiamata API TagResource per il client del producer. Consultare i log per ulteriori informazioni sulla causa principale.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000046	STATUS_INVALID_CREATE_DEVICE_RESPONSE	Creazione del dispositivo/producer non riuscita.	Gli SDK di livello superiore (ad esempio, C++ o Java) non implementano ancora l'API di creazione del dispositivo o del produttore. I client che utilizzano o direttamente il PIC possono indicare un errore tramite la notifica sui risultati.
0x52000047	STATUS_ACK_TIMESTAMP_NOT_IN_VIEW_WINDOW	Il timestamp dell'ACK di ricezione non si trova nella visualizzazione.	Questo errore si verifica se il fotogramma corrispondente all'ACK di ricezione esce dalla finestra della visualizzazione dei contenuti. In genere, ciò si verifica se la consegna dell'ACK è lenta. Può essere interpretato come un avviso e un'indicazione delle prestazioni lente del downlink.
0x52000048	STATUS_INVALID_FRAGMENT_ACK_VERSION	Versione della struttura FragmentAck non valida.	Specificare la versione corrente corretta della struttura FragmentAck.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000049	STATUS_INVALID_TOKEN_EXPIRATION	Scadenza del token di sicurezza non valida.	La scadenza del token di sicurezza dovrebbe avere un timestamp assoluto nel futuro superiore a quello attuale, con un periodo di grazia. Per i limiti relativi al periodo di tolleranza, consulta Limiti di Producer SDK .
0x5200004a	STATUS_END_OF_STREAM	Indicatore di fine del flusso (EOS).	Nel chiamata dell'API <code>GetStreamData</code> , indica che la sessione di gestione dell'upload attuale è terminata. Ciò si verifica se la sessione termina o produce degli errori oppure se il token della sessione è scaduto e la sessione viene ruotata.
0x5200004b	STATUS_DUPLICATE_STREAM_NAME	Nome di flusso duplicato.	Più flussi non possono avere lo stesso nome. Scegliere un nome univoco per il flusso.
0x5200004c	STATUS_INVALID_RETENTION_PERIOD	Periodo di conservazione non valido.	Un periodo di conservazione non valido è stato specificato nella struttura <code>StreamInfo</code> . Per informazioni sull'intervallo valido di valori per il periodo di conservazione, consulta Limiti di Producer SDK .

Codice	Messaggio	Descrizione	Azione consigliata
0x5200004d	STATUS_IN VALID_ACK _KEY_START	FragmentAck non valido.	Impossibile analizzare e la stringa ACK del frammento. Indicatore di inizio della chiave non valido. La stringa ACK del frammento potrebbe essere danneggiata. Può correggersi autonomam ente e questo errore può essere trattato come un avviso.
0x5200004e	STATUS_IN VALID_ACK _DUPLICAT E_KEY_NAME	FragmentAck non valido.	Impossibile analizzare e la stringa ACK del frammento. Più chiavi hanno lo stesso nome. La stringa ACK del frammento potrebbe essere danneggiata. Può correggersi autonomam ente e questo errore può essere trattato come un avviso.

Codice	Messaggio	Descrizione	Azione consigliata
0x5200004f	STATUS_IN VALID_ACK _INVALID_ VALUE_START	FragmentAck non valido.	Impossibile analizzare e la stringa ACK del frammento a causa di un indicatore di inizio del valore di chiave non valido. La stringa ACK del frammento potrebbe essere danneggiata. Può correggersi autonomam ente e questo errore può essere trattato come un avviso.
0x52000050	STATUS_IN VALID_ACK _INVALID_ VALUE_END	FragmentAck non valido.	Impossibile analizzare e la stringa ACK del frammento a causa di un indicatore di fine del valore di chiave non valido. La stringa ACK del frammento potrebbe essere danneggiata. Può correggersi autonomam ente e questo errore può essere trattato come un avviso.
0x52000051	STATUS_IN VALID_PAR SED_ACK_TYPE	FragmentAck non valido.	Impossibile analizzare e la stringa ACK del frammento perché è stato specificato un tipo di ACK non valido.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000052	STATUS_STREAM_HAS_BEEN_STOPPED	Il flusso è stato interrotto.	Il flusso è stato interrotto, ma un fotogramma viene comunque inserito nel flusso.
0x52000053	STATUS_INVALID_STREAM_METRICS_VERSION	Versione della struttura <code>StreamMetrics</code> non valida.	Specificare la versione corrente corretta della struttura <code>StreamMetrics</code> .
0x52000054	STATUS_INVALID_CLIENT_METRICS_VERSION	Versione della struttura <code>ClientMetrics</code> non valida.	Specificare la versione corrente corretta della struttura <code>ClientMetrics</code> .
0x52000055	STATUS_INVALID_CLIENT_READY_STATE	L'inizializzazione del producer non ha raggiunto uno stato <code>READY</code> .	Impossibile raggiungere lo stato <code>READY</code> durante l'inizializzazione del client del producer. Consultare i log per ulteriori informazioni.
0x52000056	STATUS_MACHINE_STATE_NOT_FOUND	Errore interno del computer di stato.	Non è un errore visibile pubblicamente.
0x52000057	STATUS_INVALID_FRAGMENT_ACK_TYPE	Un tipo di <code>ACK</code> non valido è stato specificato nella struttura <code>FragmentAck</code> .	La struttura <code>FragmentAck</code> deve contenere i tipi di <code>ACK</code> definiti nell'istanza pubblica.
0x52000058	STATUS_INVALID_STREAM_READY_STATE	Errore di transizione interno del computer di stato.	Non è un errore visibile pubblicamente.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000059	STATUS_CLIENT_FREE_BEFORE_STREAM	L'oggetto di flusso è stato liberato in seguito alla liberazione del producer.	Si è tentato di liberare un oggetto di flusso dopo la liberazione dell'oggetto del producer. Ciò può verificarsi soltanto nei client che utilizzano direttamente il PIC.
0x5200005a	STATUS_ALLOCATION_SIZE_SMALLER_THAN_REQUESTED	Errore interno di storage.	Un errore interno che indica che la dimensione e di allocazione effettiva dall'archivio di contenuti è inferiore alla dimensione del frame e del frammento impacchettati.
0x5200005b	STATUS_VIEW_ITEM_SIZE_GREATER_THAN_ALLOCATION	Errore interno di storage.	La dimensione archiviata dell'allocazione nella vista dei contenuti è superiore alla dimensione di allocazione nello store dei contenuti.
0x5200005c	STATUS_ACK_ERROR_STREAM_READ_ERROR	ACK di errore di lettura del flusso.	Un errore restituito dall'ACK dal backend che indica un errore di lettura o analisi dello stream. Ciò in genere si verifica quando il back-end non è in grado di recuperare e il flusso. In genere è possibile correggere questo errore tramite il ripristino automatico dello streaming.

Codice	Messaggio	Descrizione	Azione consigliata
0x5200005d	STATUS_ACK_ERR_FRAGMENT_SIZE_REACHED	La dimensione e massima del frammento è stata raggiunta.	La dimensione massima del frammento in byte è definita in Limiti di Producer SDK . Questo errore indica che sono presenti fotogrammi di dimensioni molto grandi o che non è presente nessun fotogramma chiave per la creazione di frammenti di dimensione e gestibile. Controllate le impostazioni dell'encoder e verificate che i keyframe vengano prodotti correttamente. Per i flussi con densità molto elevata, configurare il codificatore sulla creazione di frammenti in durate più ridotte per gestire la dimensione massima.

Codice	Messaggio	Descrizione	Azione consigliata
0x5200005e	STATUS_ACK_ERR_FRAGMENT_DURATION_REACHED	La durata massima del frammento è stata raggiunta.	La durata massima del frammento viene definita in Limiti di Producer SDK . Questo errore indica che sono presenti fotogrammi molto lenti al secondo o che non è presente nessun fotogramma chiave per la creazione di frammenti di durata gestibile. Controllate le impostazioni dell'encoder e verificate che i keyframe vengano prodotti correttamente a intervalli regolari.
0x5200005f	STATUS_ACK_ERR_CONNECTION_DURATION_REACHED	La durata massima della connessione è stata raggiunta.	Kinesis Video Streams impone la durata massima della connessione come specificato in Limiti di Producer SDK . L'SDK Producer ruota automaticamente lo stream o il token prima che venga raggiunto il massimo. I client che utilizzano l'SDK non dovrebbero ricevere questo errore.
0x52000060	STATUS_ACK_ERR_FRAGMENT_TIMESTAMP_NOT_MONOTONIC	I timecode non aumentano in maniera monotona.	L'SDK Producer impone i timestamp, quindi i client che utilizzano l'SDK non dovrebbero ricevere questo errore.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000061	STATUS_AC K_ERR_MUL TI_TRACK_MKV	Tracce multiple in MKV.	L'SDK Producer impone flussi a traccia singola, quindi i client che utilizzano o l'SDK non dovrebbero ricevere questo errore.
0x52000062	STATUS_AC K_ERR_INV ALID_MKV_DATA	Dati MKV non validi.	Il parser MKV di back-end ha rilevato un errore durante l'analisi del flusso. I client che utilizzano l'SDK potrebbero riscontrare questo errore se lo stream viene danneggiato durante la transizione. Ciò può verificarsi anche se le pressioni del buffer costringono l'SDK a eliminare i frame di coda che vengono parzialmente trasmessi. In quest'ultimo caso, ti consigliamo di ridurre l'FPS e la risoluzione, aumentare il rapporto di compressione o (in caso di rete «a intermittenza») aumentare la durata dell'archivio dei contenuti e del buffer per far fronte alle pressioni temporanee.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000063	STATUS_AC K_ERR_INV ALID_PROD UCER_TIMESTAMP	Timestamp del producer non valido.	Il servizio restituisce questo ACK di errore se nell'orologio del producer è presente un elevato scostamento nel futuro. Gli SDK di livello più elevato (come Java o C++) utilizzano alcune versioni dell'orologio di sistema per soddisfare il callback dell'ora corrente dal PIC. Verificate che l'orologio di sistema sia impostato correttamente. I client che utilizzano o direttamente il PIC devono verificare che le loro funzioni di callback restituiscano il timestamp corretto.
0x52000064	STATUS_AC K_ERR_STR EAM_NOT_ACTIVE	Flusso non attivo.	È stata effettuata una chiamata all'API di back-end mentre il flusso non era in stato "Attivo". Ciò si verifica quando il client crea il flusso e continua immediatamente a inserirvi i fotogrammi. L'SDK gestisce questo scenario attraverso il computer di stato e il meccanismo di recupero.

Codice	Messaggio	Descrizione	Azione consigliata
0x52000065	STATUS_AK_ERR_KMS_KEY_ACCESS_DENIED	AWS KMS errore di accesso negato.	Restituito quando l'account non dispone dell'accesso alla chiave specificata.
0x52000066	STATUS_AK_ERR_KMS_KEY_DISABLED	AWS KMS la chiave è disabilitata.	La chiave specificata è stata disabilitata.
0x52000067	STATUS_AK_ERR_KMS_KEY_VALIDATION_ERROR	AWS KMS errore di convalida della chiave.	Errore di convalida generico. Per ulteriori informazioni, consulta la Documentazione di riferimento delle API di AWS Key Management Service .
0x52000068	STATUS_AK_ERR_KMS_KEY_UNAVAILABLE	AWS KMS key non disponibile.	La chiave non è disponibile. Per ulteriori informazioni, consulta la Documentazione di riferimento delle API di AWS Key Management Service .
0x52000069	STATUS_AK_ERR_KMS_KEY_INVALID_USAGE	Uso non valido della chiave KMS.	Non AWS KMS key è configurato per essere utilizzato in questo contesto. Per ulteriori informazioni, consulta la Documentazione di riferimento delle API di AWS Key Management Service .

Codice	Messaggio	Descrizione	Azione consigliata
0x5200006a	STATUS_AK_ERR_KMS_KEY_INVALID_STATE	AWS KMS stato non valido.	Per ulteriori informazioni, consulta la Documentazione di riferimento delle API di AWS Key Management Service .
0x5200006b	STATUS_AK_ERR_KMS_KEY_NOT_FOUND	Chiave KMS non trovata.	La chiave non è stata trovata. Per ulteriori informazioni, consulta la Documentazione di riferimento delle API di AWS Key Management Service .
0x5200006c	STATUS_AK_ERR_STREAM_DELETED	Il flusso è stato eliminato o è in fase di eliminazione.	Il flusso è in fase di eliminazione da un'altra applicazione o tramite AWS Management Console.
0x5200006d	STATUS_AK_ERR_ACK_INTERNAL_ERROR	Errore interno.	Errore interno generico del servizio.
0x5200006e	STATUS_AK_ERR_FRAGMENT_ARCHIVAL_ERROR	Errore di archiviazione del frammento.	Restituito quando il servizio non riesce a mantenere in modo durevole e indicizzare il frammento. Anche se è raro, questo errore può verificarsi per diversi motivi. Per impostazione predefinita, l'SDK riprova a inviare il frammento.

Codice	Messaggio	Descrizione	Azione consigliata
0x5200006f	STATUS_ACK_ERR_UNKNOWN_ACK_ERROR	Errore sconosciuto.	Il servizio ha restituito un errore sconosciuto.
0x52000070	STATUS_MISSING_ERR_ACK_ID	Informazioni ACK mancanti.	Il parser ACK ha completato l'analisi, ma l'informazione FragmentAck è mancante.
0x52000071	STATUS_INVALID_ACK_SEGMENT_LEN	Lunghezza del segmento ACK non valida.	Nel parser ACK è stata specificata una stringa di segmento ACK con lunghezza non valida. Per ulteriori informazioni, consulta Limiti di Producer SDK .
0x52000074	STATUS_MAXIMUM_FRAGMENT_METADATA_COUNT	Il numero massimo di elementi metadati è stato aggiunto a un frammento.	Uno stream video Kinesis può aggiungere fino a 10 elementi di metadati a un frammento, aggiungendo un elemento non persistente a un frammento o aggiungendo un elemento persistente alla coda dei metadati. Per ulteriori informazioni, consulta Utilizzo dei metadati di streaming con Kinesis Video Streams .

Codice	Messaggio	Descrizione	Azione consigliata
0x52000075	STATUS_ACK_ERR_FRAGMENT_METADATA_LIMIT_REACHED	È stato raggiunto un limite (numero massimo di metadati, lunghezza del nome o del valore dei metadati)	Producer SDK limita il numero e la dimensione degli elementi metadati. Questo errore non si verifica a meno che i limiti nel codice Producer SDK non vengano modificati. Per ulteriori informazioni, consulta Utilizzo dei metadati di streaming con Kinesis Video Streams .
0x52000076	STATUS_BLOCKING_INTERRUPTED_STREAM_TERMINATED	Non ancora disponibile.	
0x52000077	STATUS_INVALID_METADATA_NAME	Il nome dei metadati non è valido.	Il nome dei metadati non può iniziare con la stringa "»AWS. Se si verifica questo errore, l'elemento di metadati non viene aggiunto al frammento o alla coda dei metadati. Per ulteriori informazioni, consulta Utilizzo dei metadati di streaming con Kinesis Video Streams .
0x52000078	STATUS_END_OF_FRAGMENT_FRAME_INVALID_STATE	Il termine di un frammento frame è in uno stato non valido.	La fine del frammento non deve essere inviata in un flusso frammentato. non-key-frame

Codice	Messaggio	Descrizione	Azione consigliata
0x52000079	STATUS_TRACK_INFO_MISSING	Le informazioni sulle tracce sono mancanti.	Il numero della traccia deve essere maggiore di zero e corrispondere all'ID della traccia.
0x5200007a	STATUS_MAXIMUM_TRACK_COUNT_EXCEEDED	È stato superato il conteggio massimo di tracce.	Puoi avere un massimo di tre tracce per stream.
0x5200007b	STATUS_OFFLINE_MODE_WITH_ZERO_RETENTION	Il tempo di conservazione della modalità di streaming offline è impostato su zero.	Il tempo di conservazione della modalità streaming offline non deve essere impostato su zero.
0x5200007c	STATUS_ACK_TRACK_NUMBER_MISMATCH	Il numero di traccia dell'errore ACK non corrisponde.	
0x5200007d	STATUS_ACK_FRAMES_MISSING_FOR_TRACK	Frame mancanti per una traccia.	
0x5200007e	STATUS_ACK_ERROR_THAN_ALLOWED_TRACKS_FOUND	Il numero massimo di tracce consentito è stato superato.	
0x5200007f	STATUS_UPLOAD_HANDLE_ABORTED	L'handle di caricamento è stata interrotta.	

Codice	Messaggio	Descrizione	Azione consigliata
0x52000080	STATUS_INVALID_CERT_PATH_LENGTH	Lunghezza percorso certificato non valida.	
0x52000081	STATUS_DUPLICATE_TRACK_ID_FOUND	Trovato ID traccia duplicato.	
0x52000082	STATUS_INVALID_CLIENT_INFO_VERSION		
0x52000083	STATUS_INVALID_CLIENT_ID_STRING_LENGTH		
0x52000084	STATUS_SETTING_KEY_FRAME_FLAG_WHILE_USING_EOFR		
0x52000085	STATUS_MAXIMUM_FRAME_TIMESTAMP_DELTA_BETWEEN_TRACKS_EXCEEDED		
0x52000086	STATUS_STREAM_SHUTTING_DOWN		

Codice	Messaggio	Descrizione	Azione consigliata
0x52000087	STATUS_CLIENT_SHUTTING_DOWN		
0x52000088	STATUS_PUT_MEDIA_LAST_PERSISTENT_ACK_NOT_RECEIVED		
0x52000089	STATUS_NON_ALIGNED_HEAP_WITHIN_CONTENT_STORE_ALLOCATORS		
0x5200008a	STATUS_MULTIPLE_CONSECUTIVE_EOFR		
0x5200008b	STATUS_DUPLICATE_STREAM_EVENT_TYPE		
0x5200008c	STATUS_STREAM_NOT_STARTED		
0x5200008d	STATUS_INVALID_IMAGE_PREFIX_LENGTH		
0x5200008e	STATUS_INVALID_METADATA_KEY_LENGTH		

Codice	Messaggio	Descrizione	Azione consigliata
0x5200008f	STATUS_INVALID_METADATA_VALUE_LENGTH		

Codici di errore e di stato restituiti dalla libreria di durata

La tabella seguente contiene informazioni sugli errori e sullo stato restituite dai metodi della Duration libreria.

Codice	Messaggio
0xFFFFFFFFFFFFFFFF	INVALID_DURATION_VALUE

Codici di errore e di stato restituiti dalla libreria comune

La tabella seguente contiene informazioni sugli errori e sullo stato restituite dai metodi della Common libreria.

Note

Le informazioni sui codici di errore e di stato sono comuni a molte API.

Codice	Codice senza 0 iniziali	Messaggio	Descrizione
0x00000001	0x1	STATUS_NULL_ARG	Valore NULL inviato per un argomento obbligatorio.
0x00000002	0x2	STATUS_INVALID_ARG	È stato specificato un valore non valido per un argomento.

Codice	Codice senza 0 iniziali	Messaggio	Descrizione
0x00000003	0x3	STATUS_INVALID_ARG_LEN	È stata specificata una lunghezza di argomento non valida.
0x00000004	0x4	STATUS_NOT_ENOUGH_MEMORY	Impossibile allocare memoria sufficiente.
0x00000005	0x5	STATUS_BUFFER_TOO_SMALL	La dimensione specificata del buffer è troppo piccola.
0x00000006	0x6	STATUS_UNEXPECTED_EOF	È stata raggiunta una fine imprevista del file.
0x00000007	0x7	STATUS_FORMAT_ERROR	È stato rilevato un formato non valido.
0x00000008	0x8	STATUS_INVALID_HANDLE_ERROR	Valore di gestione non valido.
0x00000009	0x9	STATUS_OPEN_FILE_FAILED	Impossibile aprire un file.
0x0000000a	0xa	STATUS_READ_FILE_FAILED	Impossibile leggere da un file.
0x0000000b	0xb	STATUS_WRITE_TO_FILE_FAILED	Impossibile scrivere su un file.

Codice	Codice senza 0 iniziali	Messaggio	Descrizione
0x0000000c	0xc	STATUS_INTERNAL_ERROR	Un errore interno che in genere non si verifica e che può indicare un bug dell'SDK o dell'API del servizio.
0x0000000d	0xd	STATUS_INVALID_OPERATION	Si è verificata un'operazione non valida o l'operazione non è consentita.
0x0000000e	0xe	STATUS_NOT_IMPLEMENTED	La funzione non viene implementata.
0x0000000f	0xf	STATUS_OPERATION_TIMED_OUT	Timeout dell'operazione.
0x00000010	0x10	STATUS_NOT_FOUND	Una risorsa obbligatoria non è stata trovata.
0x00000011	0x11	STATUS_CREATE_THREAD_FAILED	Impossibile creare un thread.
0x00000012	0x12	STATUS_THREAD_READ_NOT_ENOUGH_RESOURCES	Risorse insufficienti per creare un altro thread oppure è stato rilevato un limite imposto dal sistema al numero di thread.

Codice	Codice senza 0 iniziali	Messaggio	Descrizione
0x00000013	0x13	STATUS_TH READ_INVA LID_ARG	Sono stati specificati attributi del thread non validi o un altro thread è già in attesa di unirsi a questo thread.
0x00000014	0x14	STATUS_TH READ_PERM ISSIONS	Nessuna autorizzazione per impostare la politica di pianificazione e i parametri specificati negli attributi del thread.
0x00000015	0x15	STATUS_TH READ_DEAD LOCKED	È stato rilevato un deadlock o il thread di unione specifica il thread chiamante.
0x00000016	0x16	STATUS_TH READ_DOES _NOT_EXIST	Impossibile trovare il thread con l'ID del thread specificato.
0x00000017	0x17	STATUS_JO IN_THREAD _FAILED	È stato restituito un errore sconosciuto o generico dall'operazione di unione del thread.
0x00000018	0x18	STATUS_WA IT_FAILED	È stato superato il tempo massimo di attesa della variabile condizionale.

Codice	Codice senza 0 iniziali	Messaggio	Descrizione
0x00000019	0x19	STATUS_CANCEL_THREAD_FAILED	È stato restituito un errore sconosciuto o generico dall'operazione di annullamento del thread.
0x0000001a	0x1a	STATUS_THREAD_NOT_JOINABLE	L'operazione di unione dei thread è richiesta su un thread non unibile.
0x0000001b	0x1b	STATUS_DETACH_THREAD_FAILED	È stato restituito un errore sconosciuto o generico dall'operazione di distacco del thread.
0x0000001c	0x1c	STATUS_THREAD_ATTR_INIT_FAILED	Impossibile inizializzare l'oggetto degli attributi del thread.
0x0000001d	0x1d	STATUS_THREAD_ATTR_SET_STACK_SIZE_FAILED	Impossibile impostare la dimensione dello stack per l'oggetto degli attributi del thread.
0x0000001e	0x1e	STATUS_MEMORY_NOT_FREED	Usato solo nei test. Indica che non tutta la memoria richiesta è stata liberata.

Codici di errore e di stato restituiti dalla libreria heap

La tabella seguente contiene informazioni sugli errori e sullo stato restituite dai metodi della Heap libreria.

Codice	Messaggio	Descrizione
0x10000001	STATUS_HEAP_FLAGS_ERROR	È stata specificata una combinazione di flag non valida.
0x10000002	STATUS_HEAP_NOT_INITIALIZED	È stata tentata un'operazione prima dell'inizializzazione dell'heap.
0x10000003	STATUS_HEAP_CORRUPTED	L'heap era danneggiato o la banda di guardia (in modalità debug) è stata sovrascritta. Il danneggiamento dell'heap potrebbe essere causato da un overflow del buffer nel codice del client.
0x10000004	STATUS_HEAP_VRAM_LIB_MISSING	La libreria in modalità utente o kernel VRAM (RAM video) non può essere caricata o è mancante. Verificare che la piattaforma sottostante supporti le allocazioni della VRAM.
0x10000005	STATUS_HEAP_VRAM_LIB_REOPEN	Impossibile aprire la libreria della VRAM.
0x10000006	STATUS_HEAP_VRAM_INIT_FUNC_SYMBOL	Impossibile caricare l'esportazione della funzione INIT.
0x10000007	STATUS_HEAP_VRAM_ALLOC_FUNC_SYMBOL	Impossibile caricare l'esportazione della funzione ALLLOC.
0x10000008	STATUS_HEAP_VRAM_FREE_FUNC_SYMBOL	Impossibile caricare l'esportazione della funzione FREE.

Codice	Messaggio	Descrizione
0x10000009	STATUS_HEAP_VRAM_LOCK_FUNC_SYMBOL	Impossibile caricare l'esportazione della funzione LOCK.
0x1000000a	STATUS_HEAP_VRAM_UNLOCK_FUNC_SYMBOL	Impossibile caricare l'esportazione della funzione UNLOCK.
0x1000000b	STATUS_HEAP_VRAM_UNINIT_FUNC_SYMBOL	Impossibile caricare l'esportazione della funzione UNINIT.
0x1000000c	STATUS_HEAP_VRAM_GETMAX_FUNC_SYMBOL	Impossibile caricare l'esportazione della funzione GETMAX.
0x1000000d	STATUS_HEAP_DIRECT_MEM_INIT	Impossibile inizializzare il pool di heap principale nell'heap ibrido.
0x1000000e	STATUS_HEAP_VRAM_INIT_FAILED	L'inizializzazione dinamica della VRAM non è riuscita.
0x1000000f	STATUS_HEAP_LIBRARY_FREE_FAILED	Impossibile annullare l'allocazione e liberare la libreria della VRAM.
0x10000010	STATUS_HEAP_VRAM_ALLOC_FAILED	L'allocazione della VRAM non è riuscita.
0x10000011	STATUS_HEAP_VRAM_FREE_FAILED	La liberazione della VRAM non è riuscita.
0x10000012	STATUS_HEAP_VRAM_MAP_FAILED	La mappatura della VRAM non è riuscita.
0x10000013	STATUS_HEAP_VRAM_UNMAP_FAILED	L'annullamento della mappatura della VRAM non è riuscito.

Codice	Messaggio	Descrizione
0x10000014	STATUS_HEAP_VRAM_UNINIT_FAILED	La deinizializzazione della VRAM non è riuscita.
0x10000015	STATUS_INVALID_ALLOCATION_SIZE	
0x10000016	STATUS_HEAP_REALLOC_ERROR	
0x10000017	STATUS_HEAP_FILE_HEAP_FILE_CORRUPT	

Codici di errore e di stato restituiti dalla libreria MKVGen

La tabella seguente contiene informazioni sugli errori e sullo stato restituite dai metodi della MKVGen libreria.

Codice	Messaggio	Descrizione/Azione consigliata
0x32000001	STATUS_MKV_INVALID_FRAME_DATA	Membri della struttura di dati Frame non validi. Assicurati che la durata, la dimensione e i dati del frame siano validi e rientrino nei limiti specificati in Limiti di Producer SDK .
0x32000002	STATUS_MKV_INVALID_FRAME_TIMESTAMP	Timestamp del fotogramma non valido. I valori del timestamp di presentazione (PTS) e del timestamp di decodifica (DTS) calcolati sono maggiori o uguali al timestamp del fotogramma iniziale del frammento. Ciò indica un potenziale riavvio della pipeline multimediale o

Codice	Messaggio	Descrizione/Azione consigliata
		un problema di stabilità del codificatore. Per informazioni sulla risoluzione dei problemi, consulta Errore "Failed to submit frame to Kinesis Video client"
0x32000003	STATUS_MKV_INVALID_CLUSTER_DURATION	È stata specificata una durata non valida del frammento. Per ulteriori informazioni, consulta Limiti di Producer SDK .
0x32000004	STATUS_MKV_INVALID_CONTENT_TYPE_LENGTH	Lunghezza della stringa del tipo di contenuto non valida. Per ulteriori informazioni, consulta Limiti di Producer SDK .
0x32000005	STATUS_MKV_NUMBER_TOO_BIG	Si è verificato un tentativo di codifica di un numero troppo grande per essere rappresentato nel formato EBML (Extensible Binary Meta Language). Questo errore non dovrebbe essere esposto ai client dell'SDK.
0x32000006	STATUS_MKV_INVALID_CODEC_ID_LENGTH	Lunghezza della stringa dell'ID codec non valida. Per ulteriori informazioni, consulta Limiti di Producer SDK .

Codice	Messaggio	Descrizione/Azione consigliata
0x32000007	STATUS_MKV_INVALID_TRACK_NAME_LENGTH	Lunghezza della stringa del nome della traccia non valida. Per ulteriori informazioni, consulta Limiti di Producer SDK .
0x32000008	STATUS_MKV_INVALID_CODEC_PRIVATE_LENGTH	Lunghezza dei dati privati del codec non valida. Per ulteriori informazioni, consulta Limiti di Producer SDK .
0x32000009	STATUS_MKV_CODEC_PRIVATE_NULL	I dati privati del codec (CPD) sono NULL, mentre la dimensione CPD è maggiore di zero.
0x3200000a	STATUS_MKV_INVALID_TIMECODE_SCALE	Valore di scala del timecode non valido. Per ulteriori informazioni, consulta Limiti di Producer SDK .
0x3200000b	STATUS_MKV_MAX_FRAME_TIMECODE	Il timecode del fotogramma è superiore al valore massimo. Per ulteriori informazioni, consulta Limiti di Producer SDK .

Codice	Messaggio	Descrizione/Azione consigliata
0x3200000c	STATUS_MKV_LARGE_FRAME_TIMECODE	Il valore massimo del timecode del fotogramma è stato raggiunto. Il formato MKV utilizza 16-bit con firma per rappresentare il timecode relativo del fotogramma all'inizio del cluster. L'errore viene generato se il timecode del frame non può essere rappresentato. Questo errore indica una selezione di scala del timecode non valida o che la durata del cluster è troppo lunga. Di conseguenza, il timecode del fotogramma supera lo spazio di 16-bit con firma.

Codice	Messaggio	Descrizione/Azione consigliata
0x3200000d	STATUS_MKV_INVALID _ANNEXB_NALU_IN_FR AME_DATA	È stato rilevato un codice di avvio Annex-B non valido. Ad esempio, il flag di adattamento Annex-B è stato specificato e il codice ha rilevato una sequenza di avvio non valida composta da più di tre zeri. Un formato Annex-B valido deve avere una sequenza di "prevenzione di emulazione" per evitare una sequenza di tre o più zeri nel bytestream. Per ulteriori informazioni, consulta la specifica MPEG. Per ulteriori informazioni su questo errore su Android, consulta STATUS_MKV_INVALID_ANNEXB_NALU_IN_FRAME_DATA (0x3200000d) error on Android .
0x3200000e	STATUS_MKV_INVALID _AVCC_NALU_IN_FRAME DATA	Pacchetto AVCC NALU non valido quando viene specificato il flag AVCC di adattamento. Verificate che il bytestream sia in un formato AVCC valido. Per ulteriori informazioni, consulta la specifica MPEG.
0x3200000f	STATUS_MKV_BOTH_ANNEXB_AND_AVCC_SPECIFIED	Sono stati specificati sia i NALU AVCC che quelli di adattamento Annex-B. Specificare soltanto un formato o nessuno dei due.

Codice	Messaggio	Descrizione/Azione consigliata
0x32000010	STATUS_MKV_INVALID _ANNEXB_NALU_IN_CPD	Formato Annex-B non valido dei CPD durante la specifica del flag Annex-B di adattamento. Verificate che il CPD sia in formato Annex-B valido. Se non lo è, rimuovi il flag di adattamento CPD Annex-B.
0x32000011	STATUS_MKV_PTS_DTS _ARE_NOT_SAME	Kinesis Video Streams impone che il PTS (timestamp di presentazione) e il DTS (timestamp di decodifica) siano gli stessi per i frame iniziali del frammento. Si tratta dei fotogrammi chiave che avviano il frammento.
0x32000012	STATUS_MKV_INVALID _H264_H265_CPD	Impossibile analizzare i dati privati del codec H264/H265.
0x32000013	STATUS_MKV_INVALID _H264_H265_SPS_WIDTH	Impossibile estrarre la larghezza dai dati privati del codec.
0x32000014	STATUS_MKV_INVALID _H264_H265_SPS_HEIGHT	Impossibile estrarre l'altezza dai dati privati del codec.
0x32000015	STATUS_MKV_INVALID _H264_H265_SPS_NALU	NALU SPS H264/H265 non valido.
0x32000016	STATUS_MKV_INVALID _BIH_CPD	Formato di intestazione delle informazioni bitmap non valido nei dati privati del codec.

Codice	Messaggio	Descrizione/Azione consigliata
0x32000017	STATUS_MKV_INVALID_HEVC_NALU_COUNT	Conteggio unità High Efficiency Video Coding (HEVC) Network Abstraction Layer (NALU) non valido.
0x32000018	STATUS_MKV_INVALID_HEVC_FORMAT	Formato HEVC non valido.
0x32000019	STATUS_MKV_HEVC_SPS_NALU_MISSING	HEVC NALUs mancanti in Sequence Parameter Set (SPS).
0x3200001a	STATUS_MKV_INVALID_HEVC_SPS_NALU_SIZE	Dimensione HEVC SPS NALU non valida.
0x3200001b	STATUS_MKV_INVALID_HEVC_SPS_CHROMA_FORMAT_IDC	IDC formato Chroma non valido.
0x3200001c	STATUS_MKV_INVALID_HEVC_SPS_RESERVED	HEVC SPS riservato non valido.
0x3200001d	STATUS_MKV_MIN_ANNEX_B_CPD_SIZE	Dimensione minima del valore beta privato del codec. AnnexBb Per H264, questo valore deve essere maggiore o uguale a 11. Per H265, questo valore deve essere maggiore o uguale a 15.
0x3200001e	STATUS_MKV_ANNEXB_CPD_MISSING_NALUS	Dati codec privato mancanti in Annex-B NALUs.
0x3200001f	STATUS_MKV_INVALID_ANNEXB_CPD_NALUS	Beta codec privato non valido in Annex-B NALUs.

Codice	Messaggio	Descrizione/Azione consigliata
0x32000020	STATUS_MKV_INVALID_TAG_NAME_LENGTH	Lunghezza del nome del tag non valida. Il valore valido è maggiore di zero e inferiore a 128.
0x32000021	STATUS_MKV_INVALID_TAG_VALUE_LENGTH	Lunghezza del valore del tag non valida. Il valore valido è maggiore di zero e minore di 256.
0x32000022	STATUS_MKV_INVALID_GENERATOR_STATE_TAGS	Tag stato generatore non validi.
0x32000023	STATUS_MKV_INVALID_AAC_CPD_SAMPLING_FREQUENCY_INDEX	Indice non valido di frequenza di campionamento dati privati codec AAC.
0x32000024	STATUS_MKV_INVALID_AAC_CPD_CHANNEL_CONFIG	Configurazione canale dati privati codec AAC non valida.
0x32000025	STATUS_MKV_INVALID_AAC_CPD	Dati privati codec AAC non validi.
0x32000026	STATUS_MKV_TRACK_INFO_NOT_FOUND	Informazioni sulle tracce non trovate.
0x32000027	STATUS_MKV_INVALID_SEGMENT_UUID	UUID segmento non valido.
0x32000028	STATUS_MKV_INVALID_TRACK_UID	UID traccia non valido.
0x32000029	STATUS_MKV_INVALID_CLIENT_ID_LENGTH	

Codice	Messaggio	Descrizione/Azione consigliata
0x3200002a	STATUS_MKV_INVALID_AMS_ACM_CPD	
0x3200002b	STATUS_MKV_MISSING_SPS_FROM_H264_CPD	
0x3200002c	STATUS_MKV_MISSING_PPS_FROM_H264_CPD	
0x3200002d	STATUS_MKV_INVALID_PARENT_TYPE	

Codici di errore e di stato restituiti dalla libreria Trace

La tabella seguente contiene informazioni sugli errori e sullo stato restituite dai metodi della `Trace` libreria.

Codice	Messaggio
0x10100001	STATUS_MIN_PROFILER_BUFFER

Codici di errore e di stato restituiti dalla libreria Utils

La tabella seguente contiene informazioni sugli errori e sullo stato restituite dai metodi della `Utils` libreria.

Codice	Messaggio
0x40000001	STATUS_INVALID_BASE64_ENCODE
0x40000002	STATUS_INVALID_BASE
0x40000003	STATUS_INVALID_DIGIT
0x40000004	STATUS_INT_OVERFLOW

Codice	Messaggio
0x40000005	STATUS_EMPTY_STRING
0x40000006	STATUS_DIRECTORY_OPEN_FAILED
0x40000007	STATUS_PATH_TOO_LONG
0x40000008	STATUS_UNKNOWN_DIR_ENTRY_TYPE
0x40000009	STATUS_REMOVE_DIRECTORY_FAILED
0x4000000a	STATUS_REMOVE_FILE_FAILED
0x4000000b	STATUS_REMOVE_LINK_FAILED
0x4000000c	STATUS_DIRECTORY_ACCESS_DENIED
0x4000000d	STATUS_DIRECTORY_MISSING_PATH
0x4000000e	STATUS_DIRECTORY_ENTRY_STAT_ERROR
0x4000000f	STATUS_STRFTIME_FAILED
0x40000010	STATUS_MAX_TIMESTAMP_FORMAT_STR_LEN_EXCEEDED
0x40000011	STATUS_UTIL_MAX_TAG_COUNT
0x40000012	STATUS_UTIL_INVALID_TAG_VERSION
0x40000013	STATUS_UTIL_TAGS_COUNT_NON_ZERO_TAGS_NULL
0x40000014	STATUS_UTIL_INVALID_TAG_NAME_LEN
0x40000015	STATUS_UTIL_INVALID_TAG_VALUE_LEN

Codice	Messaggio
0x4000002a	STATUS_EXPONENTIAL_BACKOFF_INVALID_STATE
0x4000002b	STATUS_EXPONENTIAL_BACKOFF_RETRIES_EXHAUSTED
0x4000002c	STATUS_THREADPOOL_MAX_COUNT
0x4000002d	STATUS_THREADPOOL_INTERNAL_ERROR
0x40100001	STATUS_HASH_KEY_NOT_PRESENT
0x40100002	STATUS_HASH_KEY_ALREADY_PRESENT
0x40100003	STATUS_HASH_ENTRY_ITERATION_ABORT
0x41000001	STATUS_BIT_READER_OUT_OF_RANGE
0x41000002	STATUS_BIT_READER_INVALID_SIZE
0x41100001	STATUS_TIMER_QUEUE_STOP_SCHEDULING
0x41100002	STATUS_INVALID_TIMER_COUNT_VALUE
0x41100003	STATUS_INVALID_TIMER_PERIOD_VALUE
0x41100004	STATUS_MAX_TIMER_COUNT_REACHED
0x41100005	STATUS_TIMER_QUEUE_SHUTDOWN
0x41200001	STATUS_SEMAPHORE_OPERATION_AFTER_SHUTDOWN
0x41200002	STATUS_SEMAPHORE_ACQUIRE_WHEN_LOCKED

Codice	Messaggio
0x41300001	STATUS_FILE_LOGGER_INDEX_FILE_INVALID_SIZE

Codici di errore e di stato restituiti dalla libreria View

La tabella seguente contiene informazioni sugli errori e sullo stato restituite dai metodi della View libreria.

Codice	Messaggio	Descrizione
0x30000001	STATUS_MIN_CONTENT_VIEW_ITEMS	È stato specificato un numero di elementi di visualizzazione dei contenuti non valido. Per ulteriori informazioni, consulta Limiti di Producer SDK .
0x30000002	STATUS_INVALID_CONTENT_VIEW_DURATION	È stata specificata una durata di visualizzazione dei contenuti non valida. Per ulteriori informazioni, consulta Limiti di Producer SDK .
0x30000003	STATUS_CONTENT_VIEW_NO_MORE_ITEMS	Si è tentato di superare la posizione principale.
0x30000004	STATUS_CONTENT_VIEW_INVALID_INDEX	È stato specificato un indice non valido.
0x30000005	STATUS_CONTENT_VIEW_INVALID_TIMESTAMP	Timestamp non valido o sovrapposto. Il timestamp di decodifica del frame deve essere maggiore o uguale al timestamp del frame precedente, più la durata del frame precedente: $\text{DTS}(n) \geq \text{DTS}(n-1)$

Codice	Messaggio	Descrizione
		<p>+ Duration(n-1)`. Questo errore spesso indica un codificatore "instabile". Il codificatore produce un incremento dei fotogrammi codificati e i relativi timestamp sono inferiori rispetto alle durate intra-frame. Oppure, il flusso è configurato sull'uso dei timestamp dell'SDK e i fotogrammi vengono inviati più velocemente rispetto alla loro durata. Per ovviare a possibili instabilità del codificatore, specificare una durata del fotogramma inferiore nella struttura <code>StreamInfo.StreamCaps</code>. Per esempio, se lo stream è di 25 FPS, la durata di ogni frame è 40 ms. Tuttavia, per gestire il «jitter» dell'encoder, consigliamo di utilizzare metà della durata del fotogramma (20 ms). Alcuni flussi richiedono un controllo più preciso sul tempo per il rilevamento degli errori.</p>
0x30000006	STATUS_INVALID_CONTENT_VIEW_LENGTH	È stata specificata una lunghezza dei dati degli elementi di visualizzazione dei contenuti non valida.

Errori e codici di stato restituiti dai PutFrame callback - C producer library

La sezione seguente contiene informazioni sugli errori e sullo stato restituite dai callback per l'PutFrameoperazione all'interno della libreria C producer.

Codice	Messaggio	Descrizione	Azione consigliata
0x15000001	STATUS_STOP_CALLBACK_CHAIN	La catena di callback è stata interrotta.	
0x15000002	STATUS_MAXIMUM_CALLBACK_CHAIN	Il limite massimo della catena callback è stato raggiunto.	
0x15000003	STATUS_INVALID_PLATFORM_CALLBACKS_VERSION	Versione della struttura PlatformCallbacks non valida.	Specificare la versione corrente corretta della struttura.
0x15000004	STATUS_INVALID_PRODUCER_CALLBACKS_VERSION	Versione della struttura ProducerCallbacks non valida.	Specificare la versione corrente corretta della struttura.
0x15000005	STATUS_INVALID_STREAM_CALLBACKS_VERSION	Versione della struttura StreamCallbacks non valida.	Specificare la versione corrente corretta della struttura.
0x15000006	STATUS_INVALID_AUTH_CALLBACKS_VERSION	Versione della struttura AuthCallbacks non valida.	Specificare la versione corrente corretta della struttura.

Codice	Messaggio	Descrizione	Azione consigliata
0x15000007	STATUS_INVALID_API_CALLBACKS_VERSION	Versione della struttura ApiCallbacks non valida.	Specificare la versione corrente corretta della struttura.
0x15000008	STATUS_INVALID_AWS_CREDENTIALS_VERSION	Versione della struttura AwsCredentials non valida.	Specificare la versione corrente corretta della struttura.
0x15000009	STATUS_MAX_REQUEST_HEADER_COUNT	Il numero massimo dei conteggi dell'intestazione della richiesta è stato raggiunto.	
0x1500000a	STATUS_MAX_REQUEST_HEADER_NAME_LEN	La lunghezza massima del nome dell'intestazione della richiesta è stata raggiunta.	
0x1500000b	STATUS_MAX_REQUEST_HEADER_VALUE_LEN	La lunghezza massima del valore dell'intestazione della richiesta è stata raggiunta.	
0x1500000c	STATUS_INVALID_API_RETURN_JSON	Restituzione JSON non valida per una chiamata API.	
0x1500000d	STATUS_CURL_INIT_FAILED	Inizializzazione curl non riuscita.	

Codice	Messaggio	Descrizione	Azione consigliata
0x1500000e	STATUS_CURL_LIBRARY_INIT_FAILED	Inizializzazione lib curl non riuscita.	
0x1500000f	STATUS_INVALID_DESCRIPTOR_RETURN_JSON	JSON di ritorno non valido per DescribeStream	
0x15000010	STATUS_HMAC_GENERATION_ERROR	Errore di generazione HMAC.	
0x15000011	STATUS_IOT_FAILED	Autorizzazione IoT non riuscita.	
0x15000012	STATUS_MAX_ALIAS_LENGTH_EXCEEDED	La lunghezza massima dell'alias del ruolo è stata raggiunta.	Specificare una lunghezza dell'alias minore.
0x15000013	STATUS_MAX_AGENT_NAME_SUFFIX_LENGTH_EXCEEDED	La lunghezza massima del suffisso del nome agente è stata raggiunta.	
0x15000014	STATUS_MAX_CUSTOMER_AGENT_LENGTH_EXCEEDED	La lunghezza massima dell'agente utente cliente è stata raggiunta.	
0x15000015	STATUS_INVALID_USER_AGENT_LENGTH	Lunghezza agente utente non valida.	

Codice	Messaggio	Descrizione	Azione consigliata
0x15000016	STATUS_INVALID_ENDPOINT_CACHING_PERIOD	Periodo di caching dell'endpoint non valido.	Specificare un periodo di caching inferiore a 24 ore.
0x15000017	STATUS_IOT_EXPIRATION_OCCURS_IN_PAST	Il timestamp di scadenza dell'IoT risale al passato.	
0x15000018	STATUS_IOT_EXPIRATION_PARSING_FAILED	L'analisi della scadenza dell'IoT non è riuscita.	
0x15000019	STATUS_DUPLICATE_PRODUCER_CALLBACK_FUNC		
0x1500001a	STATUS_DUPLICATE_STREAM_CALLBACK_FREE_FUNC		
0x1500001b	STATUS_DUPLICATE_AUTH_CALLBACK_FREE_FUNC		
0x1500001c	STATUS_DUPLICATE_API_CALLBACK_FREE_FUNC		

Codice	Messaggio	Descrizione	Azione consigliata
0x1500001d	STATUS_FILE_LOGGER_INDEX_FILE_TOO_LARGE		
0x1500001e	STATUS_MAXIMUM_IOT_THING_NAME_LENGTH		
0x1500001f	STATUS_IOT_CREATE_CONTENT_FAILED		
0x15000020	STATUS_INVALID_CERT_PATH		
0x15000022	STATUS_CREDENTIAL_PROVIDER_OPEN_FILE_FAILED		
0x15000023	STATUS_CREDENTIAL_PROVIDER_INVALID_FILE_LENGTH		
0x15000024	STATUS_CREDENTIAL_PROVIDER_INVALID_FILE_FORMAT		

Codice	Messaggio	Descrizione	Azione consigliata
0x15000026	STATUS_STREAM_BEING_SHUTDOWN		
0x15000027	STATUS_CLIENT_BEING_SHUTDOWN		
0x15000028	STATUS_CONTINUOUS_RETRY_RESET_FAILED		
0x16000001	STATUS_CURL_PERFORMANCE_FAILED		
0x16000002	STATUS_IOT_INVALID_RESPONSE_LENGTH		
0x16000003	STATUS_IOT_NULL_AWS_CREDS		
0x16000004	STATUS_IOT_INVALID_URI_LEN		
0x16000005	STATUS_TIMESTAMP_STRING_UNRECOGNIZED_FORMAT		

Documentazione di riferimento per i flag di adattamento del Network Abstraction Layer (NAL)

Questa sezione contiene informazioni sui flag disponibili per l'enumerazione `StreamInfo.NalAdaptationFlags`.

Il [flusso semplice](#) in un'applicazione può essere nel formato Annex-B o AVCC :

- Il formato Annex-B delimita le [NALU \(Network Abstraction Layer Unit\)](#) con due byte di zeri, seguiti da uno o tre byte di zeri e quindi dal numero 1 (denominato codice di avvio, ad esempio 00000001).
- Anche il formato AVCC esegue il wrapping delle NALU, ma ogni NALU è preceduta da un valore che ne indica la dimensione (di solito quattro byte).

Molti codificatori producono il formato Annex-B di bitstream. Alcuni processori bitstream di livello superiore (come un motore di riproduzione o il lettore [Media Source Extensions \(MSE\)](#)) utilizzano il formato AVCC per i propri frame. AWS Management Console

Anche i dati privati del codec (CPD), che corrispondono agli SPS/PPS (Sequence Parameter Set/ Picture Parameter Set) per il codec H.264, possono essere nel formato Annex-B o AVCC. Tuttavia, per i CPD, i formati sono diversi da quelli descritti in precedenza.

I flag informano l'SDK di adattare le NALU al formato AVCC o Annex-B per i dati del fotogramma e i CPD come segue:

Flag	Adattamento
NAL_ADAPTATION_FLAG_NONE	Nessun adattamento.
NAL_ADAPTATION_ANNEXB_NALS	Adattare i NALU Annex-B ai NALU AVCC.
NAL_ADAPTATION_AVCC_NALS	Adattare i NALU AVCC ai NALU di cui all'allegato B.

Flag	Adattamento	
NAL_ADAPTATION_ANN EXB_CPD_NALS	Adattare i NALU Annex-B per i dati privati del codec ai NALU in formato AVCC.	
NAL_ADAPTATION_ANN EXB_CPD_AND_FRAME_ NALS	Adatta i NALU Annex-B per il codec e inquadra i dati privati in formato AVCC NALU.	

Per ulteriori informazioni sui tipi di NALU, consulta la sezione 1.3 relativa ai tipi di Network Abstraction Layer Unit in [RFC 3984](#).

Strutture SDK di Producer

Questa sezione include informazioni sulle strutture che è possibile utilizzare per fornire dati all'oggetto Kinesis Video Streams Producer.

Argomenti

- [DeviceInfo/DefaultDeviceInfoProvider](#)
- [StorageInfo](#)

DeviceInfo/DefaultDeviceInfoProvider

DefaultDeviceInfoProvider Gli oggetti DeviceInfo and controllano il comportamento dell'oggetto Kinesis Video Streams Producer.

Campi membri

- **versione:** un valore intero utilizzato per assicurarsi che la versione corretta della struttura venga utilizzata con la versione corrente del codice base. La versione corrente viene specificata con la macro `DEVICE_INFO_CURRENT_VERSION`.
- **name:** il nome leggibile dall'uomo per il dispositivo.
- **TagCount/tags:** attualmente non utilizzato.
- **StreamCount:** il numero massimo di stream che il dispositivo può gestire. Questo campo effettua inizialmente la pre-allocazione dello storage per i puntatori verso gli oggetti di flusso, ma gli oggetti

di flusso effettivi vengono creati in un secondo momento. Il valore predefinito è 16 flussi, ma questo valore può essere modificato nel file `DefaultDeviceInfoProvider.cpp`.

- `storageInfo`: un oggetto che descrive la configurazione principale dello storage. Per ulteriori informazioni, consulta [StorageInfo](#).

StorageInfo

Specifica la configurazione dello storage principale per Kinesis Video Streams.

L'implementazione predefinita si basa su un'implementazione di heap rapida a bassa frammentazione, ottimizzata per lo streaming. Utilizza l'allocatore `MEMALLOC`, che è possibile sovrascrivere su determinate piattaforme. Alcune piattaforme dispongono di un'allocazione di memoria virtuale senza il supporto dell'allocazione con pagine fisiche. Man mano che la memoria viene utilizzata, le pagine virtuali vengono supportate dalle pagine fisiche. Di conseguenza, quando lo storage è sottoutilizzato, la memoria viene utilizzata in modo ridotto in tutto il sistema.

Calcolare la dimensione predefinita dello storage in base alla seguente formula.

`DefragmentationFactor` deve essere impostato su 1.2 (20%).

```
Size = NumberOfStreams * AverageFrameSize * FramesPerSecond * BufferDurationInSeconds *
DefragmentationFactor
```

Nell'esempio seguente, un dispositivo dispone di flussi video e audio. Il flusso audio contiene 512 campioni al secondo, con una media di campionamento di 100 byte. Il flusso video è composto da 25 fotogrammi al secondo, con una media di campionamento di 10.000 byte. Ogni flusso ha 3 minuti di durata di buffer.

```
Size = (512 * 100 * (3 * 60) + 25 * 10000 * (3 * 60)) * 1.2 = (9216000 + 45000000) *
1.2 = 65059200 = ~ 66MB.
```

Se il dispositivo ha più memoria disponibile, ti consigliamo di aggiungere altra memoria allo storage per evitare una grave frammentazione.

Verifica che le dimensioni di archiviazione siano adeguate a contenere tutti i buffer per tutti i flussi a elevata complessità di codifica (quando la dimensione del frame è maggiore a causa dell'elevato movimento) o quando la larghezza di banda è ridotta. Se il produttore raggiunge la pressione della memoria, emette callback della pressione di overflow dello storage (`()`).

`StorageOverflowPressureFunc` Tuttavia, quando non è disponibile alcuna memoria nell'archivio dei contenuti, il frame che viene inviato a Kinesis Video Streams viene eliminato con un errore (`STATUS_STORE_OUT_OF_MEMORY = 0x5200002e`). Per ulteriori informazioni, consulta [Codici di errore e di stato restituiti dalla libreria client](#). Ciò può verificarsi anche se le ACK dell'applicazione non sono disponibili o se le ACK persistenti vengono ritardate. In questo caso, i buffer raggiungono la capacità di «durata del buffer» prima che i frame precedenti inizino a cadere.

Campi per i membri

- `versione`: un valore intero utilizzato per assicurarsi che la versione corretta della struttura venga utilizzata con la versione corrente del codice base.
- `StorageType`: enumerazione che specifica il `DEVICE_STORAGE_TYPE` supporto e l'implementazione sottostanti dello storage. Attualmente, l'unico valore supportato è `DEVICE_STORAGE_TYPE_IN_MEM`. `DEVICE_STORAGE_TYPE_HYBRID_FILE` sarà supportato in un'implementazione futura per indicare che lo storage utilizza lo store dei contenuti supportato da file.
- `StorageSize`: la dimensione di archiviazione in byte da preallocare. L'allocazione minima è 10 MB, mentre quella massima è 10 GB (questo valore cambierà nell'implementazione futura dello store dei contenuti supportato da file).
- `SpillRatio` — Un valore intero che rappresenta la percentuale di storage da allocare dal tipo di storage a memoria diretta (RAM), rispetto allo storage overflow secondario (archiviazione di file). Attualmente non utilizzato.
- `rootDirectory`: il percorso alla directory in cui è si trova lo store dei contenuti supportato da file. Attualmente non utilizzato.

Strutture dei flussi video Kinesis

È possibile utilizzare le seguenti strutture per fornire dati a un'istanza di un flusso video Kinesis.

Argomenti

- [StreamDefinition/StreamInfo](#)
- [ClientMetrics](#)
- [StreamMetrics](#)

StreamDefinition/StreamInfo

L'oggetto `StreamDefinition` nel livello C++ racchiude l'oggetto `StreamInfo` nel Platform Independent Code (PIC) e fornisce alcuni valori predefiniti nel costruttore.

Campi membri

Campo	Tipo di dati	Descrizione	Valore predefinito
<code>stream_name</code>	<code>string</code>	Nome di flusso facoltativo. Consulta Limiti di Producer SDK per ulteriori informazioni sulla lunghezza del nome del flusso. Ogni flusso deve avere un nome univoco.	Se non viene specificato alcun nome, ne verrà generato uno casualmente.
<code>retention_period</code>	<code>duration<uint64_t, ratio<3600>></code>	Il periodo di conservazione del flusso in secondi. La specifica di <code>0</code> indica nessuna conservazione.	3600 (un'ora)
<code>tags</code>	<code>const map<string, string>*</code>	Una mappa di coppie chiave-valore contenente informazioni sugli utenti. Se il flusso dispone già un set di tag, i nuovi tag verranno aggiunti al set di tag esistente.	Nessun tag
<code>kms_key_id</code>	<code>string</code>	L'ID della AWS KMS chiave da utilizzare per crittografare lo	La chiave di KMS predefinita (<code>aws/kinesis-video</code>).

Campo	Tipo di dati	Descrizione	Valore predefinito
		stream. Per ulteriori informazioni, consulta Protezione dei dati in Kinesis Video Streams .	
streaming_type	Enumerazione STREAMING_TYPE	L'unico valore supportato è STREAMING_TYPE_REALTIME .	
content_type	string	Il formato del contenuto del flusso. La console Kinesis Video Streams può riprodurre contenuti video/h264 in questo formato.	video/h264
max_latency	duration< uint64_t, milli>	La latenza massima del flusso in millisecondi. Il callback di utilizzo della latenza del flusso (se specificato) verrà richiamato se la durata del buffer supera questo intervallo di tempo. Un valore 0 indica che non verrà richiamato o nessun callback di utilizzo della latenza del flusso.	milliseconds::zero()

Campo	Tipo di dati	Descrizione	Valore predefinito
fragment_duration	duration<uint64_t>	La durata del frammento desiderata, in secondi. Tale valore viene utilizzato in combinazione con il valore key_frame_fragmentation . Se questo valore è false, Kinesis Video Streams genera frammenti su un fotogramma chiave allo scadere di tale periodo. Ad esempio, ciascun fotogramma del flusso audio Advanced Audio Coding (AAC) è un fotogramma chiave. Se viene specificato il valore key_frame_fragmentation = false , la frammentazione verrà effettuata su un fotogramma chiave in seguito a questa durata, generando frammenti di 2 secondi.	2

Campo	Tipo di dati	Descrizione	Valore predefinito
timecode_scale	<code>duration<uint64_t, milli></code>	La dimensione del timecode MKV in millisecondi, che specifica la granularità dei timecode dei fotogrammi all'interno del cluster MKV. Il timecode del frammento MKV è sempre relativo all'inizio del cluster. MKV utilizza un valore a 16-bit con firma (0-32767) per rappresentare il timecode all'interno del cluster (frammento). Verifica che il codice temporale del frame possa essere rappresentato con la scala del codice di tempo specificata. Il valore di scala predefinito del timecode di 1 ms assicura che il fotogramma di maggiori dimensioni che può essere rappresentato sia pari a 32767 ms ~ = 32 secondi. Questo valore è superiore	1

Campo	Tipo di dati	Descrizione	Valore predefinito
		alla durata massima del frammento specificata in Quote di servizio Kinesis Video Streams , pari a 10 secondi.	
key_frame_fragmentation	bool	Per scegliere se creare frammenti su un fotogramma chiave. Se true, l'SDK crea un inizio del frammento ogni volta che è presente un fotogramma chiave. Se false, Kinesis Video Streams attende <code>fragment_duration</code> almeno e produce un nuovo frammento sul fotogramma chiave successivo.	true

Campo	Tipo di dati	Descrizione	Valore predefinito
frame_timecodes	bool	Per scegliere se utilizzare i timecode del fotogramma o generare timestamp tramite il callback dell'ora corrente. Molti codificatori non creano timestamp con i fotogrammi. Quindi, specificando <code>false</code> questo parametro, si assicura che i fotogrammi abbiano il timestamp quando vengono inseriti in Kinesis Video Streams.	<code>true</code>

Campo	Tipo di dati	Descrizione	Valore predefinito
<code>absolute_fragment_times</code>	<code>bool</code>	Kinesis Video Streams utilizza MKV come meccanismo di packaging di base. La specifica MKV è rigorosa per quanto riguarda l'impostazione dei timecode del fotogramma come relativi all'inizio del cluster (frammento), ma i timecode del cluster possono essere assoluti o relativi all'ora di inizio del flusso. Se i timestamp sono relativi, la chiamata API del servizio PutMedia utilizza il timestamp di inizio facoltativo del flusso e modifica i timestamp del cluster. Il servizio archivia sempre i frammenti con i relativi timestamp assoluti.	<code>true</code>
<code>fragment_acks</code>	<code>bool</code>	Se ricevere ACK (riconoscimenti) con frammenti a livello di applicazione.	<code>true</code> , per indicare che l'SDK riceverà le ACK e agirà di conseguenza.

Campo	Tipo di dati	Descrizione	Valore predefinito
<code>restart_on_error</code>	<code>bool</code>	Per scegliere se riavviare in seguito a errori specifici.	<code>true</code> , per indicare che l'SDK tenta di riavviare lo streaming in caso di errori.
<code>recalculate_metrics</code>	<code>bool</code>	Per scegliere se calcolare nuovamente i parametri. Ogni chiamata effettuata per recuperare i parametri è in grado di ricalcolarli per ottenere il valore "in esecuzione" più recente (ciò potrebbe avere un leggero impatto sulla CPU). Sui dispositivi layout componente/con potenza estremamente ridotta, potrebbe essere necessario configurare questa impostazione su <code>false</code> per risparmiare cicli della CPU. Altrimenti, non è consigliabile utilizzare lo <code>false</code> per questo valore.	<code>true</code>

Campo	Tipo di dati	Descrizione	Valore predefinito
<code>nal_adaptation_flags</code>	<code>uint32_t</code>	Specifica i flag di adattamento dell'unità Network Abstraction Layer (NALU). Se il bitstream dispone della codifica H.264, può essere elaborato come raw o inserito in pacchetti nelle NALU, in formato Annex-B o AVCC. La maggior parte dei produttori e consumatori di stream elementari (codificatori e decodificatori di lettura) utilizza il formato Annex-B perché presenta vantaggi, come il ripristino degli errori. I sistemi di livello superiore utilizzano il formato AVCC che è il formato di default per MPEG, HLS, DASH e così via. La funzione di riproduzione della console utilizza le estensioni di sorgente multimediali (MSE) del browser per decodificare e riprodurre i flussi in formato AVCC.	L'impostazione predefinita consiste nell'adattare il formato Annex-B al formato AVCC per i dati dei fotogrammi e per i dati privati del codec.

Campo	Tipo di dati	Descrizione	Valore predefinito
		<p>Per H.264 (e per M-JPEG e H.265), nell'SDK sono presenti funzionalità di adattamento.</p> <p>Molti flussi semplici sono nel formato seguente. In questo esempio, Ab è il codice di avvio Annex-B (001 o 0001).</p> <div data-bbox="829 842 1149 1234" style="border: 1px solid #ccc; border-radius: 10px; padding: 10px; margin: 10px 0;"> <pre>Ab(Sps)Ab (Pps)Ab(I- frame)Ab(P/B- frame) Ab(P/B-fr ame)... Ab(Sps)Ab (Pps)Ab(I- frame)Ab(P/B- frame) Ab(P/B-fr ame)</pre> </div> <p>Nel caso di H.264, i dati privati del codec (CPD) rientrano nei parametri SPS (sequence parameter set) e PPS (picture parameters set) e possono essere adattati al formato AVCC. A meno che la pipeline multimediale non fornisca i</p>	

Campo	Tipo di dati	Descrizione	Valore predefinito
		<p>CPD separatamente, l'applicazione è in grado di estrarli dal fotogramma. Può farlo cercando il primo frame IDR (che dovrebbe contenere SPS e PPS), estrarre i due NALU (che lo sono) <code>Ab(Sps)</code> <code>Ab(Pps)</code>) e impostarli nel CPD in <code>StreamDefinition</code>.</p> <p>Per ulteriori informazioni, consulta Flag di adattamento del NAL.</p>	
<code>frame_rate</code>	<code>uint32_t</code>	<p>La frequenza di fotogrammi prevista. Tale valore viene utilizzato per calcolare meglio le esigenze di buffering.</p>	25
<code>avg_bandwidth_bps</code>	<code>uint32_t</code>	<p>La larghezza di banda media prevista del flusso. Tale valore viene utilizzato per calcolare meglio le esigenze di buffering.</p>	$4 * 1024 * 1024$

Campo	Tipo di dati	Descrizione	Valore predefinito
buffer_duration	duration<uint64_t>	La durata del buffer del flusso in secondi. L'SDK mantiene i frame nell'archivio dei contenuti per un massimo di <code>buffer_duration</code> , dopodiché i frame precedenti vengono eliminati man mano che la finestra avanza. Se il frame che viene eliminato non è stato inviato al backend, viene chiamato il callback del frame eliminato. Se la durata del buffer corrente è maggiore di <code>max_latency</code> , viene richiamato il callback di utilizzo della latenza del flusso. Alla ricezione dell'ACK relativa al frammento persistente, il buffer verrà tagliato in corrispondenza dell'inizio del frammento successivo per indicare che i contenuti sono stati resi persistenti in	120

Campo	Tipo di dati	Descrizione	Valore predefinito
		modo durevole nel cloud e che quindi non è più necessario archiviare i contenuti sul dispositivo locale.	

Campo	Tipo di dati	Descrizione	Valore predefinito
replay_duration	duration<uint64_t>	La durata, in secondi, necessaria per riportare il lettore corrente all'indietro per riprodurlo durante un errore, se il riavvio è abilitato. Il rollback viene interrotto all'inizio del buffer (se ha appena avviato lo streaming o se viene visualizzata l'ACK persistente). Il rollback tenta di raggiungere un fotogramma chiave che indica l'inizio di un frammento. Se l'errore che causa il riavvio non è indicativo di un host morto (l'host è ancora attivo e contiene i dati del frame nei suoi buffer interni), il rollback si interrompe all'ultimo frame ACK ricevuto. e verrà spostato al fotogramma chiave successivo, poiché l'intero frammento è già memorizzato nella memoria dell'host.	40

Campo	Tipo di dati	Descrizione	Valore predefinito
connection_staleness	duration<uint64_t>	Il tempo, in secondi, dopo il quale viene chiamata la callback stream staleness se l'SDK non riceve l'ACK di buffering. Indica che i frame vengono inviati dal dispositivo, ma il backend non li riconosce. indicativo di una connessione interrotta nel passaggio intermedio o sul sistema di bilanciamento del carico.	30
codec_id	string	L'ID codec della traccia MKV.	V_MPEG4/ISO/AVC
track_name	string	Il nome della traccia MKV.	kinesis_video

Campo	Tipo di dati	Descrizione	Valore predefinito
codecPrivateData	unsigned char*	Il buffer dei dati privati del codec (CPD). Se sulla pipeline multimediale sono presenti informazioni sui CPD prima dell'inizio del flusso, sarà possibile impostarlo su <code>StreamDefinition.codecPrivateData</code> . I bit vengono copiati e il buffer può essere riutilizzato o liberato in seguito alla chiamata per la creazione del flusso. Tuttavia, se i dati non sono disponibili al momento della creazione dello stream, possono essere impostati in uno degli overload della funzione <code>KinesisVideoStream.start(cpd)</code> .	null
codecPrivateDataDimensioni	uint32_t	La dimensione del buffer dei dati privati del codec.	0

ClientMetrics

L'ClientMetricsoggetto viene riempito chiamandogetKinesisVideoMetrics.

Campi membri

Campo	Tipo di dati	Descrizione
versione	UINT32	La versione della struttura , definita nella macro CLIENT_METRICS_CURRENT_VERSION .
contentStoreSize	UINT64	La dimensione totale dello store dei contenuti in byte. Si tratta del valore specificato in DeviceInfo.StorageInfo.storageSize .
contentStoreAvailableDimensioni	UINT64	Dimensione di archiviazione attualmente disponibile in byte.
contentStoreAllocatedDimensioni	UINT64	Dimensione attualmente allocata. La dimensione allocata e quella disponibile devono essere leggermente inferiori alla dimensione totale di storage, per via della contabilità interna e dell'implementazione dello store dei contenuti.
totalContentViewsDimensioni	UINT64	La dimensione della memoria allocata per tutte le visualizzazioni dei contenuti per tutti i flussi. Questo valore non viene conteggiato nella dimensione e dello spazio di archiviaz

Campo	Tipo di dati	Descrizione
		ione. Questa memoria viene allocata tramite la macro MEMALLOC, che è possibile sovrascrivere per fornire un allocatore personalizzato.
totalFrameRate	UINT64	La frequenza fotogrammi totale osservata su tutti i flussi.
totalTransferRate	UINT64	La velocità di streaming totale osservata in byte al secondo su tutti i flussi.

StreamMetrics

L'`StreamMetrics` oggetto viene riempito chiamando `getKinesisVideoMetrics`.

Campi membri

Campo	Tipo di dati	Descrizione
versione	UINT32	La versione della struttura , definita nella macro <code>STREAM_METRICS_CURRENT_VERSION</code> .
currentViewDuration	UINT64	La durata dei fotogrammi accumulati. Nel caso di rete veloce, questa durata è pari a zero o alla durata del frame (mentre il frame viene trasmesso). Se la durata diventa più lunga di quella <code>max_latency</code> specificata in <code>StreamDefinition</code> , viene chiamata la richiamat

Campo	Tipo di dati	Descrizione
		a della latenza del flusso, se specificata. La durata viene specificata in unità di 100ns, ovvero l'unità di tempo predefinita del livello PIC.
overallViewDuration	UINT64	La durata di visualizzazione generale. Se lo stream è configurato senza ACK o persistenza, questo valore aumenta man mano che i frame vengono inseriti nel flusso video Kinesis e diventa uguale a quello <code>buffer_duration</code> in <code>StreamDefinition</code> . Quando gli ACK sono abilitati e viene ricevuto l'ACK persistente, il buffer viene ridotto al keyframe successivo. Questo perché il timestamp ACK indica l'inizio dell'intero frammento. La durata viene specificata in unità di 100 ns, ovvero l'unità di tempo predefinita del livello PIC.
currentViewSize	UINT64	La dimensione in byte del buffer corrente.
overallViewSize	UINT64	La dimensione totale della visualizzazione in byte.
currentFrameRate	UINT64	La frequenza fotogramma osservata del flusso corrente.

Campo	Tipo di dati	Descrizione
currentTransferRate	UINT64	La velocità di trasferimento osservata in byte al secondo del flusso corrente.

Richiamate SDK di Producer

Le classi e i metodi dell'SDK Amazon Kinesis Video Streams Producer non mantengono processi propri. Vengono, invece, utilizzate le chiamate di funzione in ingresso e gli eventi per programmare callback e comunicare con l'applicazione.

Sono disponibili due modelli di callback che l'applicazione può utilizzare per interagire con l'SDK:

- [CallbackProvider](#)— Questo oggetto espone ogni callback dal componente PIC (Platform-Independent Code) all'applicazione. Questo modello consente di eseguire questa funzionalità, ma anche significa che l'implementazione deve gestire tutti i metodi API pubblici e le firme nel livello C++.
- [StreamCallbackProvider](#) e [ClientCallbackProvider](#) — Questi oggetti espongono i callback specifici dello stream e del client, mentre il livello C++ dell'SDK espone il resto dei callback. Questo è il modello di callback preferito per interagire con l'SDK Producer.

Il seguente diagramma mostra il modello di oggetti callback:

Nel diagramma precedente, `DefaultCallbackProvider` deriva da `CallbackProvider` (che espone tutti i callback nel PIC) e contiene `StreamCallbackProvider` e `ClientCallbackProvider`.

Questo argomento contiene le sezioni seguenti:

- [ClientCallbackProvider](#)
- [StreamCallbackProvider](#)
- [ClientCallbacks struttura](#)
- [Implementazioni di callback per riprovare lo streaming](#)

ClientCallbackProvider

L'oggetto `ClientCallbackProvider` espone le funzioni di callback a livello di client. I dettagli delle funzioni sono descritti nella sezione [ClientCallbacks](#).

Metodi di callback:

- `getClientReadyCallback`— Indica lo stato di disponibilità per il cliente.
- `getStorageOverflowPressureCallback`— Segnala il sovraccarico o la pressione dello storage. Questo callback viene chiamato quando l'utilizzo dello storage è inferiore al valore `STORAGE_PRESSURE_NOTIFICATION_THRESHOLD`, pari al 5% delle dimensioni dello spazio di archiviazione generale. Per ulteriori informazioni, consulta [StorageInfo](#).

StreamCallbackProvider

L'oggetto `StreamCallbackProvider` espone le funzioni di callback a livello di flusso.

Metodi di callback:

- `getDroppedFragmentReportCallback`: segnala un frammento interrotto.
- `getDroppedFrameReportCallback`— Segnala un frame perso.
- `getFragmentAckReceivedCallback`— Segnala che è stato ricevuto un frammento ACK per lo stream.
- `getStreamClosedCallback`— Segnala una condizione di chiusura dello stream.
- `getStreamConnectionStaleCallback`— Segnala una condizione di connessione obsoleta. In questa condizione, il produttore invia dati al servizio ma non riceve riconoscimenti.
- `getStreamDataAvailableCallback`— Segnala che i dati sono disponibili nello stream.
- `getStreamErrorReportCallback`— Segnala una condizione di errore dello stream.
- `getStreamLatencyPressureCallback`— Segnala una condizione di latenza del flusso, ovvero quando la dimensione del buffer accumulato è maggiore del valore. `max_latency` Per ulteriori informazioni, consulta [StreamDefinition/StreamInfo](#).
- `getStreamReadyCallback`: —Segnala una condizione di disponibilità dello stream.
- `getStreamUnderflowReportCallback`— Segnala una condizione di sottoflusso del flusso. Questa funzione non è attualmente utilizzata ed è riservata per utilizzi futuri.

Per il codice sorgente di `StreamCallbackProvider`, vedere [StreamCallbackProvider.h](#).

ClientCallbacks struttura

La struttura `ClientCallbacks` contiene i punti di accesso della funzione di callback che il PIC chiama quando si verificano determinati eventi. La struttura, inoltre, contiene le informazioni sulla versione nel campo `CALLBACKS_CURRENT_VERSION` e un campo `customData` per i dati definiti dall'utente che vengono restituiti con le singole funzioni di callback.

L'applicazione client può utilizzare un puntatore `this` per il campo `custom_data` per mappare le funzioni membro per le funzioni statiche `ClientCallback` in fase di runtime, come mostra il seguente codice di esempio:

```
STATUS TestStreamCallbackProvider::streamClosedHandler(UINT64 custom_data,
STREAM_HANDLE stream_handle, UINT64 stream_upload_handle) {
    LOG_INFO("Reporting stream stopped.");

    TestStreamCallbackProvider* streamCallbackProvider =
        reinterpret_cast<TestStreamCallbackProvider*> (custom_data);
    streamCallbackProvider->streamClosedHandler(...);
```

Eventi

Funzione	Descrizione	Type
<code>CreateDeviceFunc</code>	Non attualmente implementato nel back-end. Questa chiamata ha esito negativo quando viene effettuata da Java o C++. Altri client eseguono l'inizializzazione specifica per piattaforma.	API di back-end
<code>CreateStreamFunc</code>	Chiamata al momento della creazione del flusso	API di back-end
<code>DescribeStreamFunc</code>	Chiamata con <code>DescribeStream</code> .	API di back-end
<code>GetStreamingEndpointFunc</code>	Chiamata con <code>GetStreamingEndpoint</code> .	API di back-end

Funzione	Descrizione	Type
GetStreamingTokenFunc	Chiamata con <code>GetStreamingToken</code> .	API di back-end
PutStreamFunc	Chiamata con <code>PutStream</code> .	API di back-end
TagResourceFunc	Chiamata con <code>TagResource</code> .	API di back-end
CreateMutexFunc	Crea un mutex per la sincronizzazione.	Sincronizzazione
FreeMutexFunc	Libera il mutex.	Sincronizzazione
LockMutexFunc	Blocca il mutex di sincronizzazione.	Sincronizzazione
TryLockMutexFunc	Tenta di bloccare il mutex. Non attualmente implementato.	Sincronizzazione
UnlockMutexFunc	Sblocca il mutex.	Sincronizzazione
ClientReadyFunc	Viene chiamato quando il client entra nello stato pronto.	Notification
DroppedFrameReportFunc	Segnala quando un fotogramma viene interrotto.	Notification
DroppedFragmentReportFunc	Segnala quando un frammento viene interrotto. Questa funzione non è attualmente utilizzata ed è riservata per utilizzi futuri.	Notification

Funzione	Descrizione	Type
<code>FragmentAckReceiveFunc</code>	Viene chiamato quando viene ricevuto un frammento ACK (buffering, ricevuto, persistente e di errore).	Notification
<code>StorageOverflowPressureFunc</code>	Chiamato quando l'utilizzo dello storage è al di sotto del valore <code>STORAGE_PRESSURE_NOTIFICATION_THRESHOLD</code> , pari al 5% delle dimensioni dello spazio di archiviazione generale.	Notification
<code>StreamClosedFunc</code>	Chiamato quando vengono trasmessi gli ultimi bit dei fotogrammi rimanenti.	Notification
<code>StreamConnectionStaleFunc</code>	Chiamato quando il flusso entra in uno stato di connessione obsoleto. In questa condizione, il produttore invia dati al servizio, ma non riceve conferme.	Notification
<code>StreamDataAvailableFunc</code>	Chiamato quando sono disponibili i dati di flusso.	Notification
<code>StreamErrorReportFunc</code>	Chiamato quando si verifica un errore di flusso. Il PIC chiude automaticamente il flusso in questa condizione.	Notification

Funzione	Descrizione	Type
<code>StreamLatencyPressureFunc</code>	Chiamato quando il flusso entra in una condizione di latenza, quando la dimensione del buffer accumulata è superiore al valore <code>max_latency</code> . Per ulteriori informazioni, consulta StreamDefinition/StreamInfo .	Notification
<code>StreamReadyFunc</code>	Viene chiamato quando il flusso entra nello stato pronto.	Notification
<code>StreamUnderflowReportFunc</code>	Questa funzione non è attualmente utilizzata ed è riservata per utilizzi futuri.	Notification
<code>DeviceCertToTokenFunc</code>	Restituisce il certificato di connessione come token.	Integrazione con la piattaforma
<code>GetCurrentTimeFunc</code>	Restituisce l'orario attuale.	Integrazione con la piattaforma
<code>GetDeviceCertificateFunc</code>	Restituisce il certificato del dispositivo. Questa funzione non è attualmente utilizzata ed è riservata per utilizzi futuri.	Integrazione con la piattaforma
<code>GetDeviceFingerprintFunc</code>	Restituisce l'impronta del dispositivo. Questa funzione non è attualmente utilizzata ed è riservata per utilizzi futuri.	Integrazione con la piattaforma
<code>GetRandomNumberFunc</code>	Restituisce un numero casuale compreso tra 0 e <code>RAND_MAX</code> .	Integrazione con la piattaforma

Funzione	Descrizione	Type
<code>GetSecurityTokenFunc</code>	Restituisce il token di sicurezza passato alle funzioni che comunicano con l'API di backend. L'implementazione è in grado di specificare i <code>AccessKeyId</code> e <code>SecretKeyId</code> serializzati e il token della sessione.	Integrazione con la piattaforma
<code>LogPrintFunc</code>	Registra una riga di testo con il tag e il livello di log. Per ulteriori informazioni, consulta <code>PlatformUtils.h</code> .	Integrazione con la piattaforma

Per le funzioni di integrazione della piattaforma nella tabella precedente, l'ultimo parametro è una struttura `ServiceCallContext`, che include i campi riportati di seguito:

- `version`: la versione della strut.
- `callAfter`: un tempo assoluto dopo il quale per chiamare la funzione.
- `timeout`: il timeout dell'operazione in unità di 100 nanosecondi.
- `customData`: un valore definito dall'utente da passare al client.
- `pAuthInfo`: le credenziali per la chiamata. Per maggiori informazioni, consultare la seguente struttura (`__AuthInfo`).

Le informazioni di autorizzazione vengono fornite utilizzando la struttura `__AuthInfo` che può essere credenziali serializzate o un token di autenticazione specifico del provider. Questa struttura include i campi riportati di seguito:

- `version`: la versione della struttura `__AuthInfo`.
- `type`: un valore `AUTH_INFO_TYPE` che definisce il tipo di credenziale (certificato o token di sicurezza).
- `data`: un array di byte contenente le informazioni di autenticazione.
- `size`: la dimensione del parametro `data`.

- `expiration`: la scadenza delle credenziali in unità di 100 nanosecondi.

Implementazioni di callback per riprovare lo streaming

Kinesis Video Producer SDK fornisce lo stato dello streaming tramite le funzioni di callback. Ti consigliamo di implementare i seguenti meccanismi di callback per risolvere eventuali problemi di rete momentanei riscontrati durante lo streaming.

- Richiamata della pressione della latenza dello stream: questo meccanismo di callback viene avviato quando l'SDK rileva una condizione di latenza del flusso. Ciò accade quando la dimensione del buffer accumulato è maggiore del valore `MAX_LATENCY`. Quando il flusso viene creato, l'applicazione di streaming imposta `MAX_LATENCY` sul valore predefinito di 60 secondi. L'implementazione tipica per questo callback è il ripristino della connessione. [Se necessario, è possibile utilizzare l'implementazione di esempio su `https://github.com/aws-labs/amazon-kinesis-video-streams-producer-sdk-cpp`](https://github.com/aws-labs/amazon-kinesis-video-streams-producer-sdk-cpp). Tieni presente che non è possibile archiviare i frame non consegnati a causa di un'interruzione della rete in uno storage secondario per il riempimento.
- Stream staleness callback: questo callback viene avviato quando il produttore può inviare dati al servizio Amazon Kinesis Data Streams (uplink) ma non è in grado di recuperare i riconoscimenti (ACK bufferizzato) indietro nel tempo (l'impostazione predefinita è 60 secondi). A seconda delle impostazioni di rete, è possibile avviare lo stream latency pressure callback o lo stream staleness callback, oppure entrambi. Analogamente all'implementazione del nuovo tentativo di callback di utilizzo della latenza del flusso, l'implementazione tipica consiste nel ripristinare la connessione e avviare una nuova connessione per lo streaming. Se necessario, è possibile utilizzare l'implementazione di esempio all'[indirizzo `https://github.com/aws-labs/amazon-kinesis-video-streams-producer-c/blob/master/src/source/ConnectionStaleStateMachine`](https://github.com/aws-labs/amazon-kinesis-video-streams-producer-c/blob/master/src/source/ConnectionStaleStateMachine).
- Stream error callback: questo callback viene avviato quando l'SDK rileva un timeout sulla connessione di rete o altri errori durante la chiamata alle chiamate al servizio API KVS.
- Dropped frame callback: questo callback viene avviato quando lo spazio di archiviazione è pieno, a causa della bassa velocità di rete o di un errore di streaming. Se la velocità della rete comporta la perdita di fotogrammi, è possibile aumentare le dimensioni di archiviazione, ridurre le dimensioni dei fotogrammi video o la frequenza dei fotogrammi in modo che corrisponda alla velocità della rete.

Libreria di analisi dei flussi video Kinesis

La libreria Kinesis Video Stream Parser è un set di strumenti che è possibile utilizzare nelle applicazioni Java per utilizzare i dati MKV in un flusso video Kinesis.

La libreria include gli strumenti seguenti:

- [StreamingMkvReader](#): questa classe legge gli elementi MKV specificati da un flusso video.
- [FragmentMetadataVisitor](#): questa classe recupera i metadati relativi ai frammenti (elementi multimediali) e alle tracce (singoli flussi di dati che contengono informazioni multimediali, ad esempio audio o sottotitoli).
- [OutputSegmentMerger](#): questa classe unisce frammenti consecutivi o blocchi in un flusso video.
- [KinesisVideoExample](#): Questa è un'applicazione di esempio che mostra come utilizzare la libreria Parser per flussi video Kinesis.

La libreria include inoltre dei test che illustrano come vengono utilizzati gli strumenti.

Procedura: utilizzo della libreria Kinesis Video Stream Parser

Questa procedura include le fasi seguenti:

- [the section called “Passaggio 1: scarica e configura il codice”](#).
- [the section called “Passaggio 2: scrivere ed esaminare il codice”](#).
- [the section called “Fase 3: Esegui e verifica il codice”](#).

Prerequisiti

È necessario disporre di quanto segue per esaminare e utilizzare la libreria Kinesis Video Stream Parser:

- Un account Amazon Web Services (AWS). Se non ne hai già uno Account AWS, consulta [the section called “Registrati per un Account AWS”](#).
- [Un ambiente di sviluppo integrato \(IDE\) Java, come Eclipse Java Neon o IntelliJ IdeaJetBrains](#) .
- Java 11, ad esempio [Amazon Corretto 11](#).

Passaggio 1: scarica e configura il codice

In questa sezione, potrai scaricare la libreria Java, testare il codice e importare il progetto nell'IDE Java.

Per i prerequisiti e altri dettagli su questa procedura, consulta [Libreria parser del flusso](#).

1. Crea una directory e clona il codice sorgente della libreria dal GitHub repository (<https://github.com/aws/amazon-kinesis-video-streams-parser-library>).

```
git clone https://github.com/aws/amazon-kinesis-video-streams-parser-library
```

2. Apri l'IDE Java che stai utilizzando (ad esempio, [Eclipse](#) o [IntelliJ IDEA](#)) e importa il progetto Apache Maven che hai scaricato:
 - In Eclipse: scegliere File, Import (Importa), Maven, Existing Maven Projects (Progetti Maven esistenti) e accedere alla cartella `kinesis-video-streams-parser-lib`.
 - In IntelliJ Idea: scegliere Import (Importa). Andare al file `pom.xml` nella radice del pacchetto scaricato.

Per ulteriori informazioni, consulta la documentazione IDE correlata.

Approfondimenti

[the section called “Passaggio 2: scrivere ed esaminare il codice”](#).

Passaggio 2: scrivere ed esaminare il codice

In questa sezione, puoi esaminare la libreria Java, testare il codice e scoprire come utilizzare gli strumenti della libreria nel codice.

La libreria Kinesis Video Stream Parser contiene i seguenti strumenti:

- [StreamingMkvReader](#)
- [FragmentMetadataVisitor](#)
- [OutputSegmentMerger](#)
- [KinesisVideoExample](#)

StreamingMkvReader

Questa classe legge gli elementi MKV specificati da un flusso in modalità senza blocchi.

L'esempio di codice seguente (da `FragmentMetadataVisitorTest`) mostra come creare e utilizzare un `StreamingMkvReader` per recuperare gli oggetti `MkvElement` da un flusso di input chiamato `inputStream`:

```
StreamingMkvReader mkvStreamReader =
    StreamingMkvReader.createDefault(new
InputStreamParserByteSource(inputStream));
while (mkvStreamReader.mightHaveNext()) {
    Optional<MkvElement> mkvElement = mkvStreamReader.nextIfAvailable();
    if (mkvElement.isPresent()) {
        mkvElement.get().accept(fragmentVisitor);
        ...
    }
}
```

FragmentMetadataVisitor

Questa classe recupera i metadati per i frammenti (elementi multimediali) e tiene traccia dei singoli flussi di dati contenenti informazioni multimediali, come dati privati del codec, larghezza dei pixel o altezza dei pixel.

L'esempio di codice seguente (dal file `FragmentMetadataVisitorTest`) illustra come utilizzare `FragmentMetadataVisitor` per recuperare i dati da un oggetto `MkvElement`:

```
FragmentMetadataVisitor fragmentVisitor = FragmentMetadataVisitor.create();
StreamingMkvReader mkvStreamReader =
    StreamingMkvReader.createDefault(new InputStreamParserByteSource(in));
int segmentCount = 0;
while(mkvStreamReader.mightHaveNext()) {
    Optional<MkvElement> mkvElement = mkvStreamReader.nextIfAvailable();
    if (mkvElement.isPresent()) {
        mkvElement.get().accept(fragmentVisitor);
        if
(MkvTypeInfoos.SIMPLEBLOCK.equals(mkvElement.get().getElementMeta-data().getTypeInfo()))
{
            MkvDataElement dataElement = (MkvDataElement) mkvElement.get();
```

```
        Frame frame =
        ((MkvValue<Frame>)dataElement.getValueCopy()).getVal();
        MkvTrackMetadata trackMetadata =
        fragmentVisitor.getMkvTrackMetadata(frame.getTrackNumber());
        assertTrackAndFragmentInfo(fragmentVisitor, frame, trackMetadata);
    }
    if
    (MkvTypeInfo.SEGMENT.equals(mkvElement.get().getElementMetadata().getTypeInfo())) {
        if (mkvElement.get() instanceof MkvEndMasterElement) {
            if (segmentCount < continuationTokens.size()) {
                Optional<String> continuationToken =
                fragmentVisitor.getContinuationToken();
                Assert.assertTrue(continuationToken.isPresent());
                Assert.assertEquals(continuationTokens.get(segmentCount),
                continuationToken.get());
            }
            segmentCount++;
        }
    }
}
}
```

L'esempio precedente mostra i seguenti modelli di codifica:

- Creare un `FragmentMetadataVisitor` per analizzare i dati e un [StreamingMkvReader](#) per inserire i dati.
- Per ogni `MkvElement` nel flusso, verificare se i relativi metadati sono di tipo `SIMPLEBLOCK`.
- Se sì, recuperare il `MkvDataElement` dal `MkvElement`.
- Recuperare il `Frame` (dati multimediali) dal `MkvDataElement`.
- Recuperare il `MkvTrackMetadata` relativo al `Frame` dal `FragmentMetadataVisitor`.
- Recuperare e verificare i dati seguenti dagli oggetti `Frame` e `MkvTrackMetadata`:
 - Il numero di traccia.
 - L'altezza dei pixel del fotogramma.
 - La larghezza dei pixel del fotogramma.
 - L'ID codec relativo al codec utilizzato per la codifica del fotogramma.
 - L'arrivo del fotogramma nell'ordine corretto. Verificate che il numero di traccia del frame precedente, se presente, sia inferiore a quello del frame corrente.

Per utilizzare `FragmentMetadataVisitor` nel progetto, inviare gli oggetti `MkvElement` al visitatore tramite il relativo metodo `accept`:

```
mkvElement.get().accept(fragmentVisitor);
```

OutputSegmentMerger

Questa classe unisce i metadati di diverse tracce del flusso in un flusso con un singolo segmento.

L'esempio di codice seguente (dal file `FragmentMetadataVisitorTest`) illustra come utilizzare `OutputSegmentMerger` per unire i metadati della traccia da un array di byte chiamato `inputBytes`:

```
FragmentMetadataVisitor fragmentVisitor = FragmentMetadataVisitor.create();

ByteArrayOutputStream outputStream = new ByteArrayOutputStream();

OutputSegmentMerger outputSegmentMerger =
    OutputSegmentMerger.createDefault(outputStream);

CompositeMkvElementVisitor compositeVisitor =
    new TestCompositeVisitor(fragmentVisitor, outputSegmentMerger);

final InputStream in = TestResourceUtil.getTestInputStream("output_get_media.mkv");

StreamingMkvReader mkvStreamReader =
    StreamingMkvReader.createDefault(new InputStreamParserByteSource(in));

while (mkvStreamReader.mightHaveNext()) {
    Optional<MkvElement> mkvElement = mkvStreamReader.nextIfAvailable();
    if (mkvElement.isPresent()) {
        mkvElement.get().accept(compositeVisitor);
        if
(MkvTypeInfoos.SIMPLEBLOCK.equals(mkvElement.get().getElementMetaData().getTypeInfo()))
        {
            MkvDataElement dataElement = (MkvDataElement) mkvElement.get();
            Frame frame = ((MkvValue<Frame>) dataElement.getValueCopy()).getVal();
            Assert.assertTrue(frame.getFrameData().limit() > 0);
            MkvTrackMetadata trackMetadata =
fragmentVisitor.getMkvTrackMetadata(frame.getTrackNumber());
            assertTrackAndFragmentInfo(fragmentVisitor, frame, trackMetadata);
        }
    }
}
```

```
}
```

L'esempio precedente mostra i seguenti modelli di codifica:

- Creare un [FragmentMetadataVisitor](#) per recuperare i metadati dal flusso.
- Creare un flusso di output per ricevere i metadati uniti.
- Creare un `OutputSegmentMerger`, tramite l'invio del `ByteArrayOutputStream`.
- Creare un `CompositeMkvElementVisitor` contenente i due visitatori.
- Creare un `InputStream` che punti al file specificato.
- Unire ciascun elemento nei dati di input nel flusso di output.

KinesisVideoExample

Questa è un'applicazione di esempio che mostra come utilizzare la libreria Parser per flussi video Kinesis.

Questa classe effettua le seguenti operazioni:

- Crea un flusso video Kinesis. Se un flusso con il nome specificato è già esistente, tale flusso verrà eliminato e ricreato.
- Chiamate [PutMedia](#) per trasmettere frammenti video allo stream video Kinesis.
- Chiamate [GetMedia](#) per lo streaming di frammenti video dal flusso video di Kinesis.
- Utilizza un [StreamingMkvReader](#) per analizzare i frammenti restituiti sul flusso e un [FragmentMetadataVisitor](#) per registrare i frammenti.

Eliminazione e nuova creazione del flusso

Il seguente esempio di codice (dal `StreamOps.java` file) elimina un determinato flusso video Kinesis:

```
//Delete the stream
amazonKinesisVideo.deleteStream(new
    DeleteStreamRequest().withStreamARN(streamInfo.get().getStreamARN()));
```

Il seguente esempio di codice (dal `StreamOps.java` file) crea un flusso video Kinesis con il nome specificato:

```
amazonKinesisVideo.createStream(new CreateStreamRequest().withStreamName(streamName)
    .withDataRetentionInHours(DATA_RETENTION_IN_HOURS)
    .withMediaType("video/h264"));
```

Chiama PutMedia

Il seguente esempio di codice (dal `PutMediaWorker.java` file) chiama [PutMedia](#) lo stream:

```
putMedia.putMedia(new PutMediaRequest().withStreamName(streamName)
    .withFragmentTimecodeType(FragmentTimecodeType.RELATIVE)
    .withProducerStartTimestamp(new Date())
    .withPayload(inputStream), new PutMediaAckResponseHandler() {
    ...
});
```

Chiama GetMedia

Il seguente esempio di codice (dal `GetMediaWorker.java` file) chiama [GetMedia](#) lo stream:

```
GetMediaResult result = videoMedia.getMedia(new
    GetMediaRequest().withStreamName(streamName).withStartSelector(startSelector));
```

Analizza il risultato GetMedia

In questa sezione viene descritto come utilizzare [StreamingMkvReader](#), [FragmentMetadataVisitor](#) e [CompositeMkvElementVisitor](#) per analizzare, salvare sul file e registrare i dati restituiti da `GetMedia`.

Leggi l'output di `with GetMedia StreamingMkvReader`

Il seguente esempio di codice (dal `GetMediaWorker.java` file) crea un file [StreamingMkvReader](#) e lo utilizza per analizzare il risultato dell'[GetMedia](#) operazione:

```
StreamingMkvReader mkvStreamReader = StreamingMkvReader.createDefault(new
    InputSteamParserByteSource(result.getPayload()));
log.info("StreamingMkvReader created for stream {}", streamName);
try {
    mkvStreamReader.apply(this.elementVisitor);
} catch (MkvElementVisitException e) {
    log.error("Exception while accepting visitor {}", e);
}
```

```
}
```

Nell'esempio di codice precedente, il [StreamingMkvReader](#) recupera gli oggetti MKVElement dal payload dei risultati GetMedia. Nella sezione successiva, gli elementi vengono inviati a un [FragmentMetadataVisitor](#).

Recupera i frammenti con FragmentMetadataVisitor

Gli esempi di codice seguenti (dai file `KinesisVideoExample.java` e `StreamingMkvReader.java`) creano un [FragmentMetadataVisitor](#). Gli oggetti MkvElement iterati dal [StreamingMkvReader](#) vengono quindi inviati al visitatore tramite il metodo `accept`.

da `KinesisVideoExample.java`:

```
FragmentMetadataVisitor fragmentMetadataVisitor = FragmentMetadataVisitor.create();
```

da `StreamingMkvReader.java`:

```
if (mkvElementOptional.isPresent()) {  
    //Apply the MkvElement to the visitor  
    mkvElementOptional.get().accept(elementVisitor);  
}
```

Registrazione degli elementi e scrittura in un file

L'esempio di codice seguente (dal file `KinesisVideoExample.java`) crea gli oggetti seguenti e li restituisce come parte del valore restituito della funzione `GetMediaProcessingArguments`:

- Un `LogVisitor` (un'estensione di `MkvElementVisitor`) che effettua la scrittura nel log di sistema.
- Un `OutputStream` che scrive i dati in entrata su un file MKV.
- Un `BufferedOutputStream` che effettua il buffer dei dati associati all'`OutputStream`.
- Un [the section called "OutputSegmentMerger"](#) che unisce gli elementi consecutivi nel risultato `GetMedia` con la stessa traccia e gli stessi dati EBML.
- Un `CompositeMkvElementVisitor` che compone il [FragmentMetadataVisitor](#) visitatore [the section called "OutputSegmentMerger"](#), e `LogVisitor` in un singolo elemento.

```
//A visitor used to log as the GetMedia stream is processed.
LogVisitor logVisitor = new LogVisitor(fragmentMetadataVisitor);

//An OutputSegmentMerger to combine multiple segments that share track and ebml
metadata into one
//mkv segment.
OutputStream fileOutputStream =
Files.newOutputStream(Paths.get("kinesis_video_example_merged_output2.mkv"),
    StandardOpenOption.WRITE, StandardOpenOption.CREATE);
BufferedOutputStream outputStream = new BufferedOutputStream(fileOutputStream);
OutputSegmentMerger outputSegmentMerger =
OutputSegmentMerger.createDefault(outputStream);

//A composite visitor to encapsulate the three visitors.
CompositeMkvElementVisitor mkvElementVisitor =
    new CompositeMkvElementVisitor(fragmentMetadataVisitor,
outputSegmentMerger, logVisitor);

return new GetMediaProcessingArguments(outputStream, logVisitor,
mkvElementVisitor);
```

Gli argomenti di elaborazione multimediale vengono quindi passati a `GetMediaWorker`, che a sua volta viene passato a `ExecutorService`, che esegue il worker su un thread separato:

```
GetMediaWorker getMediaWorker = GetMediaWorker.create(getRegion(),
    getCredentialsProvider(),
    getStreamName(),
    new StartSelector().withStartSelectorType(StartSelectorType.EARLIEST),
    amazonKinesisVideo,
    getMediaProcessingArgumentsLocal.getMkvElementVisitor());
executorService.submit(getMediaWorker);
```

Approfondimenti

[the section called “Fase 3: Esegui e verifica il codice”](#)

Fase 3: Esegui e verifica il codice

La libreria Kinesis Video Stream Parser contiene strumenti pensati per essere utilizzati nei propri progetti. Il progetto contiene test di unità degli strumenti che puoi eseguire per verificare l'installazione.

Nella libreria sono inclusi i seguenti test di unità:

- `mkv`
 - `ElementSizeAndOffsetVisitorTest`
 - `MkvValueTest`
 - `StreamingMkvReaderTest`
- `utilities`
 - `FragmentMetadataVisitorTest`
 - `OutputSegmentMergerTest`

Esempi di Amazon Kinesis Video Streams

I seguenti esempi di codice mostrano come lavorare con l'API Kinesis Video Streams:

Esempi: invio di dati a Kinesis Video Streams

- [Esempio: plug-in GStreamer SDK di Kinesis Video Streams Producer - kvssink](#): mostra come creare l'SDK di Kinesis Video Streams Producer da utilizzare come destinazione GStreamer.
- [Esegui l'elemento GStreamer in un contenitore Docker](#): mostra come utilizzare un'immagine Docker predefinita per inviare video RTSP (Real-Time Streaming Protocol) da una telecamera IP a Kinesis Video Streams.
- [Esempio: streaming da una fonte RTSP](#): mostra come creare un'immagine Docker personalizzata e inviare video RTSP da una telecamera IP a Kinesis Video Streams.
- [Esempio: invio di dati a Kinesis Video PutMedia Streams utilizzando l'API](#): mostra come utilizzare l'API [Utilizzo di Java Producer Library](#) per inviare dati a Kinesis Video Streams già in formato contenitore (MKV). [PutMedia](#)

Esempi: recupero di dati da Kinesis Video Streams

- [KinesisVideoExample](#): mostra come analizzare e registrare i frammenti video utilizzando la libreria Kinesis Video Streams Parser.
- [Esempio: analisi e rendering di frammenti di Kinesis Video Streams: mostra come analizzare e renderizzare i frammenti del flusso video Kinesis utilizzando JCodec e JFrame.](#)

Esempi: riproduzione di dati video

- [Esempio: usa HLS in HTML e JavaScript](#): mostra come recuperare una sessione di streaming HLS per uno streaming video Kinesis e riprodurla in una pagina Web.

Prerequisiti

- Nel codice di esempio, fornite le credenziali specificando un profilo impostato nel file di profilo delle AWS credenziali o fornendo le credenziali nelle proprietà del sistema Java del vostro ambiente di

sviluppo integrato (IDE). Se non l'hai già fatto, imposta innanzitutto le tue credenziali. Per ulteriori informazioni, consulta [Configurare AWS le credenziali e la regione per lo sviluppo](#).

- Per visualizzare ed eseguire il codice, ti consigliamo di utilizzare un IDE Java, come uno dei seguenti:
 - [Eclipse Java Neon](#)
 - [JetBrains IntelliJ IDEA](#)

Esempio: plug-in GStreamer SDK di Kinesis Video Streams Producer - kvssink

Questo argomento descrive come creare l'SDK Amazon Kinesis Video Streams Producer da utilizzare come plug-in GStreamer.

Argomenti

- [Scarica, crea e configura l'elemento GStreamer](#)
- [Esegui l'elemento GStreamer](#)
- [Esempi di comandi di avvio di GStreamer](#)
- [Esegui l'elemento GStreamer in un contenitore Docker](#)
- [Riferimento ai parametri dell'elemento GStreamer](#)

[GStreamer](#) è un popolare framework multimediale utilizzato da più telecamere e sorgenti video per creare pipeline multimediali personalizzate combinando plugin modulari. Il plugin Kinesis Video Streams GStreamer semplifica l'integrazione della pipeline multimediale GStreamer esistente con Kinesis Video Streams. Dopo aver integrato GStreamer, puoi trasmettere video da una webcam o da una videocamera Real Time Streaming Protocol (RTSP) a Kinesis Video Streams per la riproduzione, l'archiviazione e ulteriori analisi in tempo reale o in un secondo momento.

Il plugin GStreamer gestisce automaticamente il trasferimento del flusso video su Kinesis Video Streams incapsulando la funzionalità fornita da Kinesis Video Streams Producer SDK in un elemento sink di GStreamer, `kvssink`. Il framework GStreamer fornisce un ambiente gestito standard per la costruzione di un flusso multimediale da un dispositivo, ad esempio una videocamera o altre sorgente video per ulteriori elaborazioni, il rendering o lo storage.

La pipeline GStreamer in genere contiene il link tra una sorgente (videocamera) e l'elemento sink (un lettore per il rendering del video o lo storage per il recupero offline). In questo esempio, si utilizza

l'elemento SDK Producer come sink o destinazione multimediale, per la fonte video (webcam o videocamera IP). L'elemento del plug-in che incapsula l'SDK invia quindi il flusso video a Kinesis Video Streams.

Questo argomento descrive come costruire una pipeline multimediale GStreamer in grado di trasmettere video da una sorgente video, come una webcam o un flusso RTSP, in genere collegata tramite fasi di codifica intermedie (utilizzando la codifica H.264) a Kinesis Video Streams. Quando il flusso video è disponibile come flusso video Kinesis, puoi utilizzare la [Kinesis Video Streams Parser Library](#) per ulteriori elaborazioni, riproduzione, archiviazione o analisi del flusso video.

Scarica, crea e configura l'elemento GStreamer

L'esempio del plugin GStreamer è incluso nell'SDK di Kinesis Video Streams C++ Producer. Per informazioni sui prerequisiti e il download di SDK consulta [Passaggio 1: scarica e configura il codice C++ Producer Library](#).

Puoi creare il sink Producer SDK GStreamer come libreria dinamica su macOS, Ubuntu, Raspberry Pi o Windows. Il plugin GStreamer si trova nella directory `build`. Per caricare questo plugin, deve essere nel tuo. `GST_PLUGIN_PATH` Esegui il comando seguente:

```
export GST_PLUGIN_PATH=`pwd`/build
```

Note

Su macOS, è possibile solo eseguire lo streaming di video da una videocamera di rete durante l'esecuzione di GStreamer in un container Docker. Lo streaming video da una videocamera USB su macOS in un container Docker non è supportato.

Esegui l'elemento GStreamer

Per eseguire GStreamer con l'elemento Kinesis Video Streams Producer SDK come sink, usa il comando. `gst-launch-1.0` Usa elementi upstream appropriati per l'uso del plugin GStreamer. Ad esempio, [v4l2src](#) per dispositivi v4l2 su sistemi Linux o [rtspsrc](#) per dispositivi RTSP. Specificare `kvssink` come sink (destinazione finale della pipeline) per inviare video all'SDK Producer.

Oltre a [fornire credenziali](#) e [fornire una regione](#), l'kvssinkelemento ha il seguente parametro obbligatorio:

- `stream-name`— Il nome della destinazione Kinesis Video Streams.

Per informazioni sui kvssink parametri opzionali consulta [Riferimento ai parametri dell'elemento GStreamer](#).

[Per le informazioni più recenti sui plugin e i parametri di GStreamer, vedi GStreamer Plugins.](#)

Puoi anche usare `gst-inspect-1.0` seguito dal nome di un elemento o plugin GStreamer per stamparne le informazioni e verificare che siano disponibili sul tuo dispositivo:

```
gst-inspect-1.0 kvssink
```

Se la compilazione kvssink fallisce o `GST_PLUGIN_PATH` non è impostato correttamente, il risultato sarà simile al seguente:

```
No such element or plugin 'kvssink'
```

Esempi di comandi di avvio di GStreamer

I seguenti esempi mostrano come usare il plugin kvssink GStreamer per lo streaming di video da diversi tipi di dispositivi.

Esempio 1: Streaming di video da una videocamera RTSP su Ubuntu

Il comando seguente crea una pipeline GStreamer su Ubuntu che esegue flussi in streaming da una videocamera di rete RTSP, utilizzando il plugin [rtspsrc](#) GStreamer:

```
gst-launch-1.0 -v rtspsrc location="rtsp://YourCameraRtspUrl" short-header=TRUE !  
rtph264depay ! h264parse ! kvssink stream-name="YourStreamName" storage-size=128
```

Esempio 2: codifica e riproduci in streaming video da una videocamera USB su Ubuntu

Il comando seguente crea una pipeline GStreamer su Ubuntu che codifica lo stream proveniente da una telecamera USB in formato H.264 e lo trasmette a Kinesis Video Streams. [Questo esempio utilizza il plugin GStreamer v4l2src.](#)

```
gst-launch-1.0 v4l2src do-timestamp=TRUE device=/dev/video0 ! videoconvert ! video/x-raw,format=I420,width=640,height=480,framerate=30/1 ! x264enc bframes=0 key-int-max=45 bitrate=500 ! video/x-h264,stream-format=avc,alignment=au,profile=baseline ! kvssink stream-name="YourStreamName" storage-size=512 access-key="YourAccessKey" secret-key="YourSecretKey" aws-region="YourAWSRegion"
```

Esempio 3: Streaming di video precodificati da una videocamera USB su Ubuntu

Il comando seguente crea una pipeline GStreamer su Ubuntu che trasmette video che la videocamera ha già codificato in formato H.264 su Kinesis Video Streams. [Questo esempio utilizza il plugin GStreamer v4l2src.](#)

```
gst-launch-1.0 v4l2src do-timestamp=TRUE device=/dev/video0 ! h264parse ! video/x-h264,stream-format=avc,alignment=au ! kvssink stream-name="plugin" storage-size=512 access-key="YourAccessKey" secret-key="YourSecretKey" aws-region="YourAWSRegion"
```

Esempio 4: streaming di video da una telecamera di rete su macOS

Il comando seguente crea una pipeline GStreamer su macOS che trasmette video a Kinesis Video Streams da una telecamera di rete. In questo esempio viene utilizzato il plugin [rtspsrc](#) GStreamer.

```
gst-launch-1.0 rtspsrc location="rtsp://YourCameraRtspUrl" short-header=TRUE ! rtph264depay ! h264parse ! video/x-h264, format=avc,alignment=au ! kvssink stream-name="YourStreamName" storage-size=512 access-key="YourAccessKey" secret-key="YourSecretKey" aws-region="YourAWSRegion"
```

Esempio 5: Streaming di video da una telecamera di rete su Windows

Il comando seguente crea una pipeline GStreamer su Windows che trasmette video a Kinesis Video Streams da una telecamera di rete. In questo esempio viene utilizzato il plugin [rtspsrc](#) GStreamer.

```
gst-launch-1.0 rtspsrc location="rtsp://YourCameraRtspUrl" short-header=TRUE ! rtph264depay ! video/x-h264, format=avc,alignment=au ! kvssink stream-name="YourStreamName" storage-size=512 access-key="YourAccessKey" secret-key="YourSecretKey" aws-region="YourAWSRegion"
```

Esempio 6: Streaming di video da una videocamera su Raspberry Pi

Il comando seguente crea una pipeline GStreamer su Raspberry Pi che trasmette video a Kinesis Video Streams. [Questo esempio utilizza il plugin GStreamer v4l2src.](#)

```
gst-launch-1.0 v4l2src do-timestamp=TRUE device=/dev/video0 ! videoconvert !
video/x-raw,format=I420,width=640,height=480,framerate=30/1 !
omxh264enc control-rate=1 target-bitrate=5120000 periodicity-
idr=45 inline-header=FALSE ! h264parse ! video/x-h264,stream-
format=avc,alignment=au,width=640,height=480,framerate=30/1,profile=baseline ! kvssink
stream-name="YourStreamName" access-key="YourAccessKey" secret-key="YourSecretKey"
aws-region="YourAWSRegion"
```

Esempio 7: Trasmetti in streaming audio e video in Raspberry Pi e Ubuntu

Scopri come [eseguire il comando gst-launch-1.0 per avviare lo streaming audio e video in Raspberry-Pi e Ubuntu](#).

Esempio 8: trasmetti in streaming audio e video da sorgenti di dispositivi in macOS

Scopri come [eseguire il comando gst-launch-1.0 per avviare lo streaming audio e video in MacOS](#).

Esempio 9: carica un file MKV che contiene sia audio che video

Scopri come [eseguire il comando gst-launch-1.0 per caricare il file MKV che contiene sia audio che video](#). Avrai bisogno di un file di test MKV con supporti codificati h.264 e AAC.

Esegui l'elemento GStreamer in un contenitore Docker

Docker è una piattaforma per sviluppare, distribuire ed eseguire le applicazioni utilizzando i contenitori. L'uso di Docker per creare la pipeline GStreamer standardizza l'ambiente operativo per Kinesis Video Streams, semplificando la creazione e l'utilizzo dell'applicazione.

Per installare e configurare Docker, consulta le seguenti informazioni:

- [Istruzioni di download di Docker](#)
- [Nozioni di base su Docker](#)

Dopo aver installato Docker, puoi scaricare l'SDK Kinesis Video Streams C++ Producer (e il plug-in GStreamer) dal registro Amazon Elastic Container utilizzando uno dei comandi forniti di seguito.

```
docker pull
```

Per eseguire GStreamer con l'elemento Kinesis Video Streams Producer SDK come sink in un contenitore Docker, procedi come segue:

Argomenti

- [Autentica il tuo client Docker](#)
- [Scaricare l'immagine Docker per Ubuntu, macOS, Windows o Raspberry Pi](#)
- [Esegui l'immagine Docker](#)

Autentica il tuo client Docker

Autentica il tuo client Docker nel registro Amazon ECR dal quale desideri estrarre la tua immagine. È necessario ottenere token di autenticazione per ogni registro utilizzato. I token sono validi per 12 ore. Per maggiori informazioni, consulta [Autorizzazioni del registro](#) nella Guida per l'utente di Amazon Elastic Container Registry.

Example : autenticazione con Amazon ECR

Per autenticarti con Amazon ECR, copia e incolla il seguente comando come illustrato.

```
sudo aws ecr get-login-password --region us-west-2 | docker login -u AWS --password-stdin https://546150905175.dkr.ecr.us-west-2.amazonaws.com
```

In caso di esito positivo, l'output stampa Login Succeeded.

Scaricare l'immagine Docker per Ubuntu, macOS, Windows o Raspberry Pi

Scaricare l'immagine Docker per l'ambiente Docker utilizzando uno dei comandi riportati di seguito, a seconda del sistema operativo in uso:

Scaricare l'immagine Docker per Ubuntu

```
sudo docker pull 546150905175.dkr.ecr.us-west-2.amazonaws.com/kinesis-video-producer-sdk-cpp-amazon-linux:latest
```

Scaricare l'immagine Docker per macOS

```
docker pull 546150905175.dkr.ecr.us-west-2.amazonaws.com/kinesis-video-producer-sdk-cpp-amazon-linux:latest
```

Scaricare l'immagine Docker per Windows

```
docker pull 546150905175.dkr.ecr.us-west-2.amazonaws.com/kinesis-video-producer-sdk-cpp-amazon-windows:latest
```

Scaricare l'immagine Docker per Raspberry Pi

```
sudo docker pull 546150905175.dkr.ecr.us-west-2.amazonaws.com/kinesis-video-producer-sdk-cpp-raspberry-pi:latest
```

Per verificare che l'immagine sia stata aggiunta, eseguire il comando riportato di seguito:

```
docker images
```

Esegui l'immagine Docker

Utilizzare uno dei seguenti comandi per eseguire l'immagine Docker, a seconda del sistema operativo in uso:

Esegui l'immagine Docker su Ubuntu

```
sudo docker run -it --network="host" --device=/dev/video0 546150905175.dkr.ecr.us-west-2.amazonaws.com/kinesis-video-producer-sdk-cpp-amazon-linux /bin/bash
```

Esegui l'immagine Docker su macOS

```
sudo docker run -it --network="host" 546150905175.dkr.ecr.us-west-2.amazonaws.com/kinesis-video-producer-sdk-cpp-amazon-linux /bin/bash
```

Esegui l'immagine Docker su Windows

```
docker run -it 546150905175.dkr.ecr.us-west-2.amazonaws.com/kinesis-video-producer-sdk-cpp-windows AWS_ACCESS_KEY_ID AWS_SECRET_ACCESS_KEY RTSP_URL STREAM_NAME
```

Esegui l'immagine Docker su Raspberry Pi

```
sudo docker run -it --device=/dev/video0 --device=/dev/vchiq -v /opt/vc:/opt/vc 546150905175.dkr.ecr.us-west-2.amazonaws.com/kinesis-video-producer-sdk-cpp-raspberry-pi /bin/bash
```

Docker avvia il contenitore e presenta un prompt dei comandi per l'utilizzo dei comandi all'interno del contenitore.

Nel container, impostare le variabili di ambiente utilizzando il comando seguente:

```
export LD_LIBRARY_PATH=/opt/awssdk/amazon-kinesis-video-streams-producer-sdk-cpp/  
kinesis-video-native-build/downloads/local/lib:$LD_LIBRARY_PATH  
export PATH=/opt/awssdk/amazon-kinesis-video-streams-producer-sdk-cpp/kinesis-video-  
native-build/downloads/local/bin:$PATH  
export GST_PLUGIN_PATH=/opt/awssdk/amazon-kinesis-video-streams-producer-sdk-cpp/  
kinesis-video-native-build/downloads/local/lib:$GST_PLUGIN_PATH
```

Inizia lo streaming per `kvssink` utilizzando il per `gst-launch-1.0` eseguire una pipeline appropriata per il tuo dispositivo e la tua sorgente video. Ad esempio, le pipeline, vedi. [Esempi di comandi di avvio di GStreamer](#)

Riferimento ai parametri dell'elemento GStreamer

Per inviare video all'SDK Amazon Kinesis Video Streams Producer C++, `kvssink` devi specificare come sink o destinazione finale della pipeline. La documentazione di riferimento fornisce informazioni sui parametri `kvssink` obbligatori e facoltativi. Per ulteriori informazioni, consulta [the section called "Plugin GStreamer - kvssink"](#).

Argomenti

- [the section called "Fornisci le credenziali a kvssink"](#)
- [the section called "Fornisci una regione a kvssink"](#)
- [the section called "kvssinkparametri opzionali"](#)

Fornisci le credenziali a **kvssink**

Per consentire all'elemento `kvssink` GStreamer di effettuare richieste AWS, fornisci AWS le credenziali da utilizzare quando chiama il servizio Amazon Kinesis Video Streams. La catena di fornitori di credenziali cerca le credenziali nel seguente ordine:

1. AWS IoT credenziali

Per impostare AWS IoT le credenziali, vedere. [the section called "Controllo dell'accesso alle risorse Kinesis Video Streams tramite AWS IoT"](#)

Il valore del `iot-credentials` parametro deve iniziare con `iot-certificate`, ed essere seguito da un elenco separato da virgole delle seguenti coppie `chiave=valore`.

Chiave	Richiesto	Descrizione
ca-path	Si	<p>Percorso del file del certificato CA utilizzato per stabilire la fiducia con il servizio di backend tramite TLS.</p> <p>Example</p> <p>Esempio: <i>/file/path/to/certificate.pem</i></p>
cert-path	Si	<p>Percorso del file del certificato X.509.</p> <p>Example</p> <p>Esempio: <i>/file/path/to/certificateID -certificate.pem.crt</i></p>
endpoint	Si	<p>L'endpoint del provider di AWS IoT Core credenziali per il tuo account. AWS Consulta la Guida per gli sviluppatori.AWS IoT</p> <p>Example</p> <p>Esempio: <i>credential-account-specific-prefix .credentials.iot. aws-region .amazonaws.com</i></p>
key-path	Si	<p>Percorso del file della chiave privata utilizzata nella coppia di chiavi pubblica/privata.</p>

Chiave	Richiesto	Descrizione
		<p>Example</p> <p>Esempio: <code>/file/path/to/certificateID -private.pem.key</code></p>
<code>role-aliases</code>	Sì	<p>Il nome dell'alias del ruolo che punta al ruolo AWS IAM da utilizzare durante la connessione a. AWS IoT Core</p> <p>Example</p> <p>Esempio: <code>KvsCameraIoRoleAlias</code></p>
<code>iot-thing-name</code>	No	<p><code>iot-thing-name</code> È facoltativo. Se non <code>iot-thing-name</code> viene fornito, viene utilizzato il valore del <code>stream-name</code> parametro.</p> <p>Example</p> <p>Esempio: <code>kvs_example_camera</code></p>

Example

Esempio:

```
gst-launch-1.0 -v ... ! kvssink stream-name="YourStream" aws-region="YourRegion"
  iot-certificate="iot-certificate,endpoint=credential-account-specific-prefix.credentials.iot.aws-region.amazonaws.com,cert-path=certificateID-certificate.pem.crt,key-path=certificateID-private.pem.key,ca-path=certificate.pem,role-aliases=YourRoleAlias,iot-thing-name=YourThingName"
```

2. Variabili di ambiente

Per `kvssink` utilizzare le credenziali dell'ambiente, impostate le seguenti variabili di ambiente:

Nome della variabile di ambiente	Richiesto	Descrizione
<code>AWS_ACCESS_KEY_ID</code>	Sì	La chiave di AWS accesso utilizzata per accedere ad Amazon Kinesis Video Streams.
<code>AWS_SECRET_ACCESS_KEY</code>	Sì	La chiave AWS segreta associata alla chiave di accesso.
<code>AWS_SESSION_TOKEN</code>	No	Specifica il valore del token di sessione richiesto se si utilizzano credenziali di sicurezza temporanee direttamente dalle AWS STS operazioni.

L'impostazione della variabile di ambiente modifica il valore utilizzato fino al termine della sessione della shell o finché non imposti la variabile su un valore diverso. Per rendere le variabili persistenti nelle sessioni future, impostale nello script di avvio della shell.

3. `access-key`, `secret-key` parametri

Per specificare le credenziali direttamente come `kvssink` parametro, impostate i seguenti parametri:

<code>kvssink</code> Nome del parametro	Richiesto	Descrizione
<code>access-key</code>	Sì	La chiave di AWS accesso utilizzata per accedere ad Amazon Kinesis Video Streams.

kvssinkNome del parametro	Richiesto	Descrizione
secret-key	Sì	La chiave AWS segreta associata alla chiave di accesso.
session-token	No	Specifica il valore del token di sessione richiesto se si utilizzano credenziali di sicurezza temporanee direttamente dalle AWS STS operazioni.

Example

Utilizzo di credenziali statiche:

```
gst-launch-1.0 -v ... ! kvssink stream-name="YourStream" aws-region="YourRegion"
  access-key="AKIDEXAMPLE" secret-key="SKEEXAMPLE"
```

Example

Utilizzo di credenziali temporanee:

```
gst-launch-1.0 -v ... ! kvssink stream-name="YourStream" aws-region="YourRegion"
  access-key="AKIDEXAMPLE" secret-key="SKEEXAMPLE" session-token="STEXAMPLE"
```

4. File di credenziali

Important

Se hai selezionato uno dei metodi precedenti, non puoi utilizzare il `credential-file` kvssink parametro.

kvssinkNome del parametro	Richiesto	Descrizione
credential-file	Sì	Percorso del file di testo contenente le credenziali in un formato specifico.

Il file di testo deve contenere credenziali in uno dei seguenti formati:

- CREDENZIALI *YourAccessKeyYourSecretKey*
- *Scadenza delle credenziali YourAccessKeyYourSecretKeySessionToken*

Example

Esempio: il *credentials.txt* file si trova in `/home/ubuntu` e contiene quanto segue:

```
CREDENTIALS AKIDEXAMPLE 2023-08-10T22:43:00Z SKEXAMPLE STEXAMPLE
```

Per utilizzarlo in kvssink, digita:

```
gst-launch-1.0 -v ... ! kvssink stream-name="YourStream" aws-region="YourRegion"
credential-file="/home/ubuntu/credentials.txt"
```

Note

Il tempo di scadenza dovrebbe essere di almeno $5 + 30 + 3 = 38$ secondi nelle future. Il periodo di grazia è definito come la `IOT_CREDENTIAL_FETCH_GRACE_PERIOD` variabile in [IotCredentialProvider.h](#). Se all'avvio le credenziali sono troppo vicine alla scadenza kvssink, viene visualizzato il codice di errore. `0x52000049 - STATUS_INVALID_TOKEN_EXPIRATION`

Important

kvssink non modifica il file delle credenziali. Se utilizzi credenziali temporanee, il file delle credenziali deve essere aggiornato da una fonte esterna prima della scadenza meno il periodo di prova.

Fornisci una regione a **kvssink**

Di seguito è riportato l'ordine di ricerca della regione:

1. `AWS_DEFAULT_REGION`La variabile di ambiente viene esaminata per prima. Se è impostata, quella regione viene utilizzata per configurare il client.
2. `aws-region`il parametro viene esaminato successivamente. Se è impostato, quella regione viene utilizzata per configurare il client.
3. Se non è stato utilizzato nessuno dei metodi precedenti, il `kvssink` valore predefinito è. `us-west-2`

kvssinkparametri opzionali

L'elemento `kvssink` ha i seguenti parametri opzionali. Per ulteriori informazioni su questi parametri, consultare [Strutture dei flussi video Kinesis](#).

Parametro	Descrizione	Unità/Tipo	Predefinita
<code>stream-name</code>	<p>Il nome dello stream video Amazon Kinesis di destinazione.</p> <div style="border: 1px solid #f08080; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p> Important</p> <p>Se non viene specificato alcun nome di stream, verrà utilizzato il nome di stream predefinito: «DEFAULT_STREAM».</p> <p>Se non esiste già uno stream con</p> </div>		

Parametro	Descrizione	Unità/Tipo	Predefinita
	<p>quel nome predefinito, verrà creato.</p>		
<code>absolute-fragment-times</code>	Per scegliere se usare i tempi assoluti del frammento.	Booleano	true
<code>access-key</code>	<p>La chiave di AWS accesso utilizzata per accedere a Kinesis Video Streams.</p> <p>Devi avere AWS le credenziali impostate o fornire questo parametro. Per fornire queste informazioni, digitare quanto segue:</p> <pre>export AWS_ACCESS_KEY_ID=</pre>		
<code>avg-bandwidth-bps</code>	La larghezza di banda media prevista del flusso.	Bit al secondo	4194304

Parametro	Descrizione	Unità/Tipo	Predefinita
<code>aws-region</code>	<p>Il Regione AWS da usare.</p> <div data-bbox="472 352 792 1335"><p> Note È inoltre possibile fornire alla regione la variabile di ambiente di <code>AWS_DEFAULT_REGION</code> ambiente. Le variabili di ambiente hanno la precedenza a se sono impostati sia la variabile di ambiente che i parametri kvssink.</p></div> <div data-bbox="472 1402 792 1772"><p> Important La regione verrà impostata di default <code>us-west-2</code> se non</p></div>	Stringa	<code>"us-west-2"</code>

Parametro	Descrizione	Unità/Tipo	Predefinita
	diversamente specificato.		
<code>buffer-duration</code>	Durata del buffer del flusso.	Secondi	120
<code>codec-id</code>	L'ID del codec del flusso.	Stringa	"V_MPEG4/ISO/AVC"
<code>connection-staleness</code>	L'ora dopo la quale viene chiamato lo stream staleness callback.	Secondi	60
<code>content-type</code>	Il tipo di contenuto del flusso.	Stringa	"video/h264"
<code>fragment-acks</code>	Per scegliere se utilizzare il frammento ACK.	Booleano	true
<code>fragment-duration</code>	La durata del frammento desiderato.	Millisecondi	2000
<code>framerate</code>	La frequenza fotogrammi prevista.	Fotogrammi al secondo	25
<code>frame-timecodes</code>	Per scegliere se utilizzare i timecode del fotogramma o generare timestamp tramite il callback dell'ora corrente.	Booleano	true

Parametro	Descrizione	Unità/Tipo	Predefinita
<code>key-frame-fragmentation</code>	Per scegliere se creare frammenti su un fotogramma chiave.	Booleano	true
<code>log-config</code>	Percorso di configurazione del log.	Stringa	"../kvs_log_configuration"
<code>max-latency</code>	La latenza massima del flusso.	Secondi	60
<code>recalculate-metrics</code>	Per scegliere se calcolare nuovamente i parametri.	Booleano	true
<code>replay-duration</code>	La durata per spostare all'indietro il lettore corrente per la riproduzione quando si verifica un errore se il riavvio è abilitato.	Secondi	40
<code>restart-on-error</code>	Per scegliere se riavviare quando si verifica un errore.	Booleano	true
<code>retention-period</code>	Il periodo di tempo in cui il flusso viene conservato.	Ore	2
<code>rotation-period</code>	Il periodo di rotazione delle chiavi. Per ulteriori informazioni, vedete AWS KMS Rotating Keys .	Secondi	3600

Parametro	Descrizione	Unità/Tipo	Predefinita
secret-key	<p>La chiave AWS segreta utilizzata per accedere a Kinesis Video Streams.</p> <p>Devi avere AWS le credenziali impostate o fornire questo parametro.</p> <pre>export AWS_SECRET_ACCESS_KEY=</pre>		
session-token	<p>Specifica il valore del token di sessione richiesto se si utilizzano credenziali di sicurezza temporane e direttamente dalle operazioni. AWS STS</p>		
storage-size	<p>La dimensione di archiviazione del dispositivo in mebibyte (MiB). Per informazioni sulla configurazione dello storage del dispositivo, consulta StorageInfo.</p>	Mebibyte (MiB)	128

Parametro	Descrizione	Unità/Tipo	Predefinita
<code>streaming-type</code>	Tipo di streaming. I valori validi includono: <ul style="list-style-type: none">• 0: tempo reale• 1: quasi in tempo reale (non attualmente supportato)• 2: non in linea	Enum <code>GstKvsSinkStreamingType</code>	0: tempo reale
<code>timecode-scale</code>	La dimensione del timecode MKV.	Millisecondi	1
<code>track-name</code>	Il nome della traccia MKV.	Stringa	"kinesis_video"

Parametro	Descrizione	Unità/Tipo	Predefinita
<code>iot-certificate</code>	<p>AWS IoT credenziali da utilizzare nell'kvssinkelemento.</p> <p><code>iot-certificate</code> accetta le seguenti chiavi e valori:</p> <div data-bbox="472 625 792 1318" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p> Note</p> <p><code>iot-thing-name</code> è facoltativo. Se non <code>iot-thing-name</code> viene fornito, viene utilizzato il valore del <code>stream-name</code> parametro.</p> </div> <ul style="list-style-type: none"> • <code>endpoint=iotcredentialsproviderendpoint</code> • <code>cert-path=/localdirectorypath/to/certificate</code> • <code>key-path=/localdir</code> 	Stringa	Nessuno

Parametro	Descrizione	Unità/Tipo	Predefinita
	ectorypath / to/private/ key <ul style="list-style-type: none"> • ca-path=/ localdir ectorypath/ to/ca-cert • role-aliases =role-aliases • iot-thing-name=YourIotThingName 		

Esempio: invio di dati a Kinesis Video PutMedia Streams utilizzando l'API

Questo esempio dimostra come utilizzare l'API. [PutMedia](#) Mostra come inviare dati già in formato contenitore (MKV). Se i dati devono essere assemblati in un formato contenitore prima dell'invio (ad esempio, se state assemblando i dati video della videocamera in fotogrammi), consultate. [Librerie per produttori di Kinesis Video Streams](#)

Note

L'PutMediaoperazione è disponibile solo negli SDK C++ e Java. Ciò è dovuto alla gestione full duplex delle connessioni, del flusso di dati e dei riconoscimenti. Non è supportato in altre lingue.

Questo esempio include le fasi seguenti:

- [Passaggio 1: scarica e configura il codice](#)
- [Passaggio 2: scrivere ed esaminare il codice](#)
- [Passaggio 3: Esegui e verifica il codice](#)

Passaggio 1: scarica e configura il codice

Segui i passaggi per scaricare il codice di esempio Java, importare il progetto nel tuo IDE Java, configurare le posizioni delle librerie e configurare il codice per utilizzare AWS le tue credenziali.

1. Crea una directory e clona il codice sorgente di esempio dal GitHub repository. L'esempio PutMedia è parte della [Java Producer Library](#).

```
git clone https://github.com/aws-labs/amazon-kinesis-video-streams-producer-sdk-java
```

2. Apri l'IDE Java che stai utilizzando (ad esempio, [Eclipse](#) o [IntelliJ IDEA](#)) e importa il progetto Apache Maven che hai scaricato:
 - In Eclipse: scegliere File, Import (Importa), Maven, Existing Maven Projects (Progetti Maven esistenti) e andare alla radice del pacchetto scaricato. Selezionare il file pom.xml.
 - In IntelliJ Idea: scegliere Import (Importa). Andare al file pom.xml nella radice del pacchetto scaricato.

Per ulteriori informazioni, consulta la documentazione IDE correlata.

3. Aggiornare il progetto per fare in modo che l'IDE possa individuare le librerie importate.
 - Per IntelliJ IDEA, procedere come segue:
 - a. Facendo clic con il pulsante destro del mouse, aprire il menu contestuale della directory lib del progetto e scegliere Add as library (Aggiungi come libreria).
 - b. Scegli File, quindi scegli Struttura del progetto.
 - c. In Project Settings (Impostazioni progetto), scegliere Modules (Moduli).
 - d. Nella scheda Sources (Origini), impostare Language Level (Livello linguaggio) su **7** o superiore.
 - Per Eclipse, procedere come segue:
 - a. Facendo clic con il pulsante destro del mouse, aprire il menu contestuale del progetto e scegliere Properties (Proprietà), Java Build Path (Percorso di compilazione Java), Source (Origine). Quindi, esegui queste operazioni:
 1. Nella scheda Source (Origine), fare doppio clic su Native library location (Percorso libreria nativa).

2. Nella procedura guidata Native Library Folder Configuration (Configurazione cartella libreria nativa), scegliere Workspace (Area di lavoro).
 3. Nel menu di selezione Native Library Folder (Cartella libreria nativa), scegliere la directory `lib` nel progetto.
- b. Facendo clic con il pulsante destro del mouse, aprire il menu contestuale del progetto e scegliere Properties (Proprietà). Quindi, esegui queste operazioni:
1. Nella scheda Libraries (Librerie), scegliere Add Jars (Aggiungi Jar).
 2. Nella procedura guidata JAR selection (Selezione JAR), scegliere tutti i file `.jar` nella directory `lib` del progetto.

Passaggio 2: scrivere ed esaminare il codice

L'esempio dell'API PutMedia (PutMediaDemo) mostra i seguenti modelli di codifica:

Argomenti

- [Crea il PutMediaClient](#)
- [Trasmetti contenuti multimediali e metti in pausa il thread](#)

Gli esempi di codice riportati in questa sezione sono presi dalla classe PutMediaDemo.

Crea il PutMediaClient

La creazione PutMediaClient dell'oggetto richiede i seguenti parametri:

- L'URI dell'endpoint PutMedia.
- Un InputStream diretto al file MKV di cui effettuare lo streaming.
- Il nome del flusso. In questo esempio viene utilizzato il flusso creato in [Utilizzo di Java Producer Library](#) (`my-stream`). Per utilizzare un flusso diverso, modificare il parametro seguente:

```
private static final String STREAM_NAME="my-stream";
```

Note

L'esempio di PutMedia API non crea uno stream. È necessario creare uno stream utilizzando l'applicazione di test per [Utilizzo di Java Producer Library](#), la console Kinesis Video Streams o il. AWS CLI

- Il timestamp corrente.
- Il tipo di timecode. L'esempio utilizza il codice RELATIVE, che indica che il timestamp è relativo all'inizio del container.
- Un oggetto `AWSKinesisVideoV4Signer` che verifica che i pacchetti ricevuti siano stati inviati da un mittente autorizzato.
- La larghezza di banda di upstream massima in Kbps.
- Un oggetto `AckConsumer` per la ricezione delle conferme di ricezione dei pacchetti.

Il codice seguente crea l'oggetto `PutMediaClient`:

```
/* actually URI to send PutMedia request */
final URI uri = URI.create(KINESIS_VIDEO_DATA_ENDPOINT + PUT_MEDIA_API);

/* input stream for sample MKV file */
final InputStream inputStream = new FileInputStream(MKV_FILE_PATH);

/* use a latch for main thread to wait for response to complete */
final CountDownLatch latch = new CountDownLatch(1);

/* a consumer for PutMedia ACK events */
final AckConsumer ackConsumer = new AckConsumer(latch);

/* client configuration used for AWS SigV4 signer */
final ClientConfiguration configuration = getClientConfiguration(uri);

/* PutMedia client */
final PutMediaClient client = PutMediaClient.builder()
    .putMediaDestinationUri(uri)
    .mkvStream(inputStream)
    .streamName(STREAM_NAME)
    .timestamp(System.currentTimeMillis())
    .fragmentTimeCodeType("RELATIVE")
    .signWith(getKinesisVideoSigner(configuration))
```

```
.upstreamKbps(MAX_BANDWIDTH_KBPS)
.receiveAcks(ackConsumer)
.build();
```

Trasmetti contenuti multimediali e metti in pausa il thread

In seguito alla creazione del client, il campione avvia lo streaming asincrono con `putMediaInBackground`. Il thread principale viene quindi messo in pausa tramite `latch.await` fino al ritorno di `AckConsumer` e a questo punto il client viene chiuso.

```
/* start streaming video in a background thread */
    client.putMediaInBackground();

    /* wait for request/response to complete */
    latch.await();

    /* close the client */
    client.close();
```

Passaggio 3: Esegui e verifica il codice

Per eseguire l'esempio dell'API `PutMedia`, procedere come segue:

1. Crea uno stream denominato `my-stream` nella console Kinesis Video Streams o utilizzando AWS CLI
2. Per modificare la directory di lavoro nella directory SDK del producer Java:

```
cd /<YOUR_FOLDER_PATH_WHERE_SDK_IS_DOWNLOADED>/amazon-kinesis-video-streams-producer-sdk-java/
```

3. Per compilare l'SDK Java e l'applicazione di demo:

```
mvn package
```

4. Creare un filename temporaneo nella directory `/tmp`:

```
jar_files=$(mktemp)
```

5. Creare una stringa di classpath di dipendenze dall'archivio locale a un file:

```
mvn -Dmdep.outputFile=$jar_files dependency:build-classpath
```

6. Impostare il valore della variabile di ambiente LD_LIBRARY_PATH come segue:

```
export LD_LIBRARY_PATH=/<YOUR_FOLDER_PATH_WHERE_SDK_IS_DOWNLOADED>/amazon-kinesis-  
video-streams-producer-sdk-cpp/kinesis-video-native-build/downloads/local/lib:  
$LD_LIBRARY_PATH  
$ classpath_values=$(cat $jar_files)
```

7. Esegui la demo dalla riga di comando come segue, fornendo le tue AWS credenziali:

```
java -classpath target/kinesisvideo-java-demo-1.0-SNAPSHOT.jar:$classpath_values -  
Daws.accessKeyId=${ACCESS_KEY} -Daws.secretKey=${SECRET_KEY} -Djava.library.path=/  
opt/amazon-kinesis-video-streams-producer-sdk-cpp/kinesis-video-native-build  
com.amazonaws.kinesisvideo.demoapp.DemoAppMain
```

8. Apri la console [Kinesis Video Streams](#) e scegli il tuo stream nella pagina Gestisci stream. Viene avviata la riproduzione del video nel riquadro Video Preview (Anteprima video).

Esempio: streaming da una fonte RTSP

[C++ Producer Library](#) Contiene una definizione per un contenitore [Docker che si connette a una telecamera](#) di rete RTSP (Real-Time Streaming Protocol). L'uso di Docker standardizza l'ambiente operativo per Kinesis Video Streams, semplificando la creazione e l'utilizzo dell'applicazione.

La procedura seguente illustra come configurare e utilizzare l'applicazione demo di RTSP.

Argomenti

- [Tutorial video](#)
- [Prerequisiti](#)
- [Crea l'immagine Docker](#)
- [Esegui l'applicazione di esempio RTSP](#)

Tutorial video

Questo video mostra come configurare un Raspberry Pi per inviare feed RTSP al AWS cloud e ad Amazon Kinesis Video Streams. Questa è una end-to-end dimostrazione.

Questo video dimostra come acquisire immagini da un feed per utilizzare la visione artificiale e Amazon Rekognition per elaborare le immagini e inviare avvisi.

Prerequisiti

Per eseguire l'applicazione di esempio RTSP Kinesis Video Streams, è necessario disporre di quanto segue:

- Docker: per informazioni sull'installazione e l'utilizzo di Docker, consulta i seguenti collegamenti:
 - [Istruzioni di download di Docker](#)
 - [Nozioni di base su Docker](#)
- Origine videocamera di rete RTSP: per informazioni sulle videocamere consigliate consulta [Requisiti di sistema](#).

Crea l'immagine Docker

Innanzitutto, crea l'immagine Docker all'interno della quale verrà eseguita l'applicazione demo.

1. Clona l'archivio demo di Amazon Kinesis Video Streams.

```
git clone https://github.com/aws-samples/amazon-kinesis-video-streams-demos.git
```

2. Passa alla directory contenente il Dockerfile. In questo caso, è la directory [docker-rtsp](#).

```
cd amazon-kinesis-video-streams-demos/producer-cpp/docker-rtsp/
```

3. Usa il seguente comando per creare l'immagine Docker. Questo comando crea l'immagine e la etichetta come `rtspdockertest`.

```
docker build -t rtspdockertest .
```

4. Esegui `docker images` e cerca l'ID dell'immagine contrassegnato con `rtspdockertest`

Ad esempio, nell'output di esempio riportato di seguito, `IMAGE ID` è `54f0d65f69b2`.

REPOSITORY	TAG	IMAGE ID	CREATED	PLATFORM	SIZE
rtspdockertest	latest	54f0d65f69b2	10 minutes ago	linux/arm64	653.1 MiB

Ne avrai bisogno in una fase successiva.

Esegui l'applicazione di esempio RTSP

È possibile eseguire l'applicazione di esempio RTSP dall'interno o dall'esterno del contenitore Docker. Segui le istruzioni appropriate riportate di seguito.

Argomenti

- [All'interno del contenitore Docker](#)
- [Fuori dal contenitore Docker](#)

All'interno del contenitore Docker

Esegui l'applicazione di esempio RTSP

1. Avvia il contenitore Docker Amazon Kinesis Video Streams utilizzando il seguente comando:

```
docker run -it YourImageId /bin/bash
```

2. Per avviare l'applicazione di esempio, fornisci AWS le tue credenziali, il nome del flusso video di Amazon Kinesis e l'URL della telecamera di rete RTSP.

Important

Se utilizzi credenziali temporanee, dovrai fornire anche le tue. `AWS_SESSION_TOKEN`
Vedi il secondo esempio riportato di seguito.

```
export AWS_ACCESS_KEY_ID=YourAccessKeyId  
export AWS_SECRET_ACCESS_KEY=YourSecretKeyId  
export AWS_DEFAULT_REGION=YourAWSRegion  
./kvs_gstreamer_sample YourStreamName YourRtspUrl
```

Credenziali temporanee:

```
export AWS_ACCESS_KEY_ID=YourAccessKeyId  
export AWS_SECRET_ACCESS_KEY=YourSecretKeyId
```

```
export AWS_SESSION_TOKEN=YourSessionToken
export AWS_DEFAULT_REGION=YourAWSRegion
./kvs_gstreamer_sample YourStreamName YourRtspUrl
```

3. Accedi AWS Management Console e apri la console [Kinesis Video Streams](#).

Visualizza lo stream.

4. Per uscire dal contenitore Docker, chiudi la finestra del terminale o digita `exit`.

Fuori dal contenitore Docker

Dall'esterno del contenitore Docker, usa il seguente comando:

```
docker run -it YourImageId /bin/bash -c "export AWS_ACCESS_KEY_ID=YourAccessKeyId;
export AWS_SECRET_ACCESS_KEY=YourSecretKeyId; export
AWS_SESSION_TOKEN=YourSessionToken; export AWS_DEFAULT_REGION=Your AWS Region; ./
kvs_gstreamer_sample YourStreamName YourRtspUrl"
```

Esempio: analisi e rendering di frammenti di Kinesis Video Streams

[Libreria parser del flusso](#) Contiene un'applicazione demo denominata

`KinesisVideoRendererExample` che dimostra l'analisi e il rendering di frammenti di flussi video di Amazon Kinesis. L'esempio utilizza [JCodec](#) per decodificare i fotogrammi con codifica H.264 acquisiti tramite l'applicazione [Esempio: plug-in GStreamer SDK di Kinesis Video Streams Producer - kvssink](#). In seguito alla decodifica del fotogramma tramite JCodec, viene effettuato il rendering dell'immagine visibile mediante [JFrame](#).

In questo esempio viene mostrato come effettuare le operazioni seguenti:

- Recupera i fotogrammi da un flusso video Kinesis utilizzando `GetMedia` l'API ed esegui il rendering del flusso per la visualizzazione.
- Visualizza il contenuto video degli stream in un'applicazione personalizzata anziché utilizzare la console Kinesis Video Streams.

Puoi anche utilizzare le classi in questo esempio per visualizzare il contenuto del flusso video Kinesis che non è codificato come H.264, come un flusso di file JPEG che non richiedono la decodifica prima di essere visualizzati.

La procedura seguente illustra come configurare e utilizzare l'applicazione demo di Renderer.

Prerequisiti

Per esaminare e utilizzare la libreria di esempio di Renderer, occorre quanto segue:

- Un account Amazon Web Services (AWS). Se non disponi già di un AWS account, consulta la sezione [Guida introduttiva a Kinesis Video Streams](#).
- [Un ambiente di sviluppo integrato \(IDE\) Java, come Eclipse Java Neon o IntelliJ IdeaJetBrains](#).

Esecuzione dell'esempio del renderer

1. Crea una directory, quindi clona il codice sorgente di esempio dal repository. GitHub

```
git clone https://github.com/aws/amazon-kinesis-video-streams-parser-library
```

2. Aprire l'IDE Java in uso (ad esempio, [Eclipse](#) o [IntelliJ IDEA](#)) e importare il progetto Apache Maven scaricato:
 - In Eclipse: scegliere File, Import (Importa), Maven, Existing Maven Projects (Progetti Maven esistenti). Passa alla directory `kinesis-video-streams-parser-lib`.
 - In IntelliJ Idea: scegliere Import (Importa). Andare al file `pom.xml` nella radice del pacchetto scaricato.

Note

Se IntelliJ non è in grado di trovare le dipendenze, potrebbe essere necessario eseguire quanto segue:

- Build clean: scegliere File (File), Settings (Impostazioni), Build, Execution, Deployment (Creazione, esecuzione, distribuzione), Compiler (Compilatore). Verificate che sia selezionata l'opzione Clear output directory on rebuild, quindi scegliete Build, Build Project.
- Reimport the project (Reimporta il progetto): aprire il menu di scelta rapida (fare clic con il pulsante destro del mouse) del progetto e scegliere Maven, Reimport (Reimporta).

Per ulteriori informazioni, consulta la documentazione IDE correlata.

3. Dall'IDE Java, aprire `src/test/java/com.amazonaws.kinesisvideo.parser/examples/KinesisVideoRendererExampleTest`.
4. Rimuovere la direttiva `@Ignore` dal file.
5. Aggiorna il `.stream` parametro con il nome del tuo stream video Kinesis.
6. Eseguire il test `KinesisVideoRendererExample`.

Come funziona

L'applicazione di esempio dimostra quanto segue:

- [Invio dei dati MKV](#)
- [Analisi dei frammenti MKV in frame](#)
- [Decodifica e visualizzazione del frame](#)

Invio dei dati MKV

L'esempio invia dati MKV di esempio dal `rendering_example_video.mkv` file, utilizzandoli `PutMedia` per inviare dati video a uno stream denominato `render-example-stream`

L'applicazione crea un `PutMediaWorker`:

```
PutMediaWorker putMediaWorker = PutMediaWorker.create(getRegion(),
    getCredentialsProvider(),
    getStreamName(),
    inputStream,
    streamOps.amazonKinesisVideo);
executorService.submit(putMediaWorker);
```

Per informazioni sulle classi `PutMediaWorker`, consulta [Chiama PutMedia](#) nella documentazione della [Libreria parser del flusso](#).

Analisi dei frammenti MKV in frame

L'esempio quindi recupera e analizza i frammenti MKV dal flusso tramite un `GetMediaWorker`:

```
GetMediaWorker getMediaWorker = GetMediaWorker.create(getRegion(),
```

```
getCredentialsProvider(),
getStreamName(),
new StartSelector().withStartSelectorType(StartSelectorType.EARLIEST),
streamOps.amazonKinesisVideo,
getMediaProcessingArgumentsLocal().getFrameVisitor());
executorService.submit(getMediaWorker);
```

Per ulteriori informazioni sulle classi `GetMediaWorker`, consulta [Chiama GetMedia](#) nella documentazione della [Libreria parser del flusso](#).

Decodifica e visualizzazione del frame

L'esempio quindi decodifica e visualizza il fotogramma tramite [JFrame](#).

Il seguente esempio di codice è tratto dalla classe `KinesisVideoFrameViewer`, che estende `JFrame`:

```
public void setImage(BufferedImage bufferedImage) {
    image = bufferedImage;
    repaint();
}
```

L'immagine viene visualizzata come un'istanza di [java.awt.image. BufferedImage](#). Per esempi che illustrano come lavorare con `BufferedImage`, consulta la sezione relativa alla [lettura/al caricamento di un'immagine](#).

Monitoraggio dei flussi video di Amazon Kinesis

Il monitoraggio è una parte importante per mantenere l'affidabilità, la disponibilità e le prestazioni di Amazon Kinesis Video Streams AWS e delle tue soluzioni. Ti consigliamo di raccogliere i dati di monitoraggio da tutte le parti della AWS soluzione per aiutarti a eseguire il debug di un errore multipunto, se si verifica. Prima di iniziare a monitorare Amazon Kinesis Video Streams, ti consigliamo di creare un piano di monitoraggio che includa le risposte alle seguenti domande:

- Quali sono gli obiettivi del monitoraggio?
- Di quali risorse si intende eseguire il monitoraggio?
- Con quale frequenza sarà eseguito il monitoraggio di queste risorse?
- Quali strumenti di monitoraggio verranno utilizzati?
- Chi eseguirà i processi di monitoraggio?
- Chi deve ricevere una notifica quando si verifica un problema?

Dopo aver definito gli obiettivi di monitoraggio e creato il piano di monitoraggio, il passaggio successivo consiste nello stabilire una linea di base per le normali prestazioni di Amazon Kinesis Video Streams nel tuo ambiente. È necessario misurare le prestazioni di Amazon Kinesis Video Streams in momenti diversi e in condizioni di carico diverse. Durante il monitoraggio di Amazon Kinesis Video Streams, memorizza una cronologia dei dati di monitoraggio che hai raccolto. Puoi confrontare le prestazioni attuali di Amazon Kinesis Video Streams con questi dati storici per aiutarti a identificare modelli di prestazioni normali e anomalie prestazionali e ideare metodi per risolvere i problemi che potrebbero sorgere.

Argomenti

- [Monitoraggio dei parametri di Amazon Kinesis Video Streams con CloudWatch](#)
- [Monitoraggio dell'agente Amazon Kinesis Video Streams Edge con CloudWatch](#)
- [Registrazione delle chiamate API Amazon Kinesis Video Streams con AWS CloudTrail](#)

Monitoraggio dei parametri di Amazon Kinesis Video Streams con CloudWatch

Puoi monitorare un flusso video Kinesis utilizzando Amazon CloudWatch, che raccoglie ed elabora i dati grezzi da Amazon Kinesis Video Streams in metriche leggibili quasi in tempo reale. Queste

statistiche vengono registrate per un periodo di 15 mesi in modo da poter accedere alle informazioni storiche e avere una prospettiva migliore sulle prestazioni della tua applicazione o del tuo servizio web.

Nella console [Amazon Kinesis Video Streams](#), CloudWatch puoi visualizzare i parametri per un flusso video Amazon Kinesis in due modi:

- Nella pagina Dashboard (Pannello di controllo) scegliere la scheda Video streams (Flussi video) nella sezione Account-level metrics for Current Region (Parametri a livello di account per la regione corrente).
- Scegliere la scheda Monitoring (Monitoraggio) nella pagina dei dettagli del flusso video.

Amazon Kinesis Video Streams fornisce le seguenti metriche:

Parametro	Descrizione
ArchivedFragmentsConsumed.Media	<p>Il numero di punti di quota multimediali relativi ai frammenti utilizzati da tutte le API. Per una spiegazione del concetto di punti di quota, vedere the section called “Framment-metadati e framment-media quote”</p> <p>Unità: numero</p>
ArchivedFragmentsConsumed.Metadata	<p>Il numero di frammenti, metadati, punti di quota utilizzati da tutte le API. Per una spiegazione del concetto di punti di quota, vedi the section called “Framment-metadati e framment-media quote”</p> <p>Unità: numero</p>
PutMedia.Requests	<p>Il numero di richieste PutMedia API per un determinato flusso.</p> <p>Unità: numero</p>
PutMedia.IncomingBytes	<p>Il numero di byte ricevuti come parte dello PutMedia stream.</p> <p>Unità: byte</p>

Parametro	Descrizione
<code>PutMedia.IncomingFragments</code>	<p>Il numero di frammenti completi ricevuti come parte dello <code>PutMedia</code> stream.</p> <p>Unità: numero</p>
<code>PutMedia.IncomingFrames</code>	<p>Il numero di frame completi ricevuti come parte dello <code>PutMedia</code> stream.</p> <p>Unità: numero</p>
<code>PutMedia.ActiveConnections</code>	<p>Il numero totale di connessioni al service host.</p> <p>Unità: numero</p>
<code>PutMedia.ConnectionErrors</code>	<p>Gli errori durante lo stabilire <code>PutMedia</code> la connessione per lo stream.</p> <p>Unità: numero</p>
<code>PutMedia.FragmentIngestionLatency</code>	<p>La differenza di tempo tra il momento in cui il primo e l'ultimo byte di un frammento vengono ricevuti da Amazon Kinesis Video Streams.</p> <p>Unità: millisecondi</p>
<code>PutMedia.FragmentPersistLatency</code>	<p>Il tempo impiegato dal momento in cui i dati completi del frammento vengono ricevuti e archiviati.</p> <p>Unità: numero</p>
<code>PutMedia.Latency</code>	<p>La differenza di tempo tra la richiesta e la risposta HTTP ricevuta <code>InletService</code> durante la creazione della connessione.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>PutMedia.BufferingAckLatency</code>	<p>La differenza di tempo tra il momento in cui il primo byte di un nuovo frammento viene ricevuto da Amazon Kinesis Video Streams e l'invio del Buffering ACK per il frammento.</p> <p>Unità: millisecondi</p>
<code>PutMedia.ReceivedAckLatency</code>	<p>La differenza di tempo tra il momento in cui l'ultimo byte di un nuovo frammento viene ricevuto da Amazon Kinesis Video Streams e l'invio dell'ACK ricevuto per il frammento.</p> <p>Unità: millisecondi</p>
<code>PutMedia.PersistedAckLatency</code>	<p>La differenza di tempo tra il momento in cui l'ultimo byte di un nuovo frammento viene ricevuto da Amazon Kinesis Video Streams e l'invio del Persisted ACK per il frammento.</p> <p>Unità: millisecondi</p>
<code>PutMedia.ErrorAckCount</code>	<p>Il numero di ACK di errore inviati durante l'esecuzione dello streaming. <code>PutMedia</code></p> <p>Unità: numero</p>
<code>PutMedia.Success</code>	<p>1 per ogni frammento scritto senza errori; 0 per ogni frammento con errori. Il valore medio di questo parametro indica il numero di frammenti completi e validi inviati.</p> <p>Unità: numero</p>
<code>GetMedia.Requests</code>	<p>Il numero di richieste <code>GetMedia</code> API per un determinato flusso.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>GetMedia.OutgoingBytes</code>	<p>Il numero totale di byte inviati dal servizio come parte dell'<code>GetMedia</code> API per un determinato flusso.</p> <p>Unità: byte</p>
<code>GetMedia.OutgoingFragments</code>	<p>Il numero di frammenti inviati durante l'esecuzione dello <code>GetMedia</code> stream.</p> <p>Unità: numero</p>
<code>GetMedia.OutgoingFrames</code>	<p>Il numero di frame inviati durante <code>GetMedia</code> lo stream specificato.</p> <p>Unità: numero</p>
<code>GetMedia.MillisBehindNow</code>	<p>La differenza di orario tra il timestamp corrente del server e il timestamp del server dell'ultimo frammento inviato.</p> <p>Unità: millisecondi</p>
<code>GetMedia.ConnectionErrors</code>	<p>Il numero di connessioni che non sono state stabilite con successo.</p> <p>Unità: numero</p>

Parametro	Descrizione
GetMedia.Success	<p>1 per ogni frammento inviato senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p>Note</p> <p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle richieste, consulta Registrazione del riepilogo delle AWS richieste e delle risposte.</p> </div> <p>Unità: numero</p>
GetMediaForFragmentList.OutgoingBytes	<p>Il numero totale di byte inviati dal servizio come parte dell'GetMediaForFragmentList API per un determinato flusso.</p> <p>Unità: byte</p>
GetMediaForFragmentList.OutgoingFragments	<p>Il numero totale di frammenti inviati dal servizio come parte dell'GetMediaForFragmentList API per un determinato flusso.</p> <p>Unità: numero</p>
GetMediaForFragmentList.OutgoingFrames	<p>Il numero totale di frame inviati dal servizio come parte dell'GetMediaForFragmentList API per un determinato flusso.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>GetMediaForFragmentList.Requests</code>	<p>Il numero di richieste <code>GetMediaForFragmentList</code> API per un determinato flusso.</p> <p>Unità: numero</p>
<code>GetMediaForFragmentList.Success</code>	<p>1 per ogni frammento inviato senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div data-bbox="748 606 1508 1016" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p> Note</p> <p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle AWS richieste, consulta Registrazione del riepilogo delle richieste e delle risposte.</p> </div> <p>Unità: numero</p>
<code>ListFragments.Latency</code>	<p>La latenza delle chiamate <code>ListFragments</code> API per il nome dello stream specificato.</p> <p>Unità: millisecondi</p>
<code>ListFragments.Requests</code>	<p>Il numero di richieste <code>ListFragments</code> API per un determinato flusso.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>ListFragments.Success</code>	<p>1 per ogni richiesta senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div data-bbox="748 352 1508 758"><p> Note</p><p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle AWS richieste, consulta Registrazione del riepilogo delle richieste e delle risposte.</p></div> <p>Unità: numero</p>
<code>GetHLSStreamingSessionURL.Latency</code>	<p>La latenza delle chiamate <code>GetHLSStreamingSessionURL</code> API per il nome dello stream specificato.</p> <p>Unità: millisecondi</p>
<code>GetHLSStreamingSessionURL.Requests</code>	<p>Il numero di richieste <code>GetHLSStreamingSessionURL</code> API per un determinato flusso.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>GetHLSStreamingSessionURL.Success</code>	<p>1 per ogni richiesta senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div><p> Note</p><p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle AWS richieste, consulta Registrazione del riepilogo delle richieste e delle risposte.</p></div> <p>Unità: numero</p>
<code>GetHLSMasterPlaylist.Latency</code>	<p>La latenza delle chiamate <code>GetHLSMasterPlaylist</code> API per il nome dello stream specificato.</p> <p>Unità: millisecondi</p>
<code>GetHLSMasterPlaylist.Requests</code>	<p>Il numero di richieste <code>GetHLSMasterPlaylist</code> API per un determinato flusso.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>GetHLSMasterPlaylist.Success</code>	<p>1 per ogni richiesta senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div data-bbox="748 352 1508 758"><p> Note</p><p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle AWS richieste, consulta Registrazione del riepilogo delle richieste e delle risposte.</p></div> <p>Unità: numero</p>
<code>GetHLSMediaPlaylist.Latency</code>	<p>La latenza delle chiamate <code>GetHLSMediaPlaylist</code> API per il nome dello stream specificato.</p> <p>Unità: millisecondi</p>
<code>GetHLSMediaPlaylist.Requests</code>	<p>Il numero di richieste <code>GetHLSMediaPlaylist</code> API per un determinato flusso.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>GetHLSMediaPlaylist.Success</code>	<p>1 per ogni richiesta senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div><p> Note</p><p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle AWS richieste, consulta Registrazione del riepilogo delle richieste e delle risposte.</p></div> <p>Unità: numero</p>
<code>GetMP4InitFragment.Latency</code>	<p>La latenza delle chiamate <code>GetMP4InitFragment</code> API per il nome dello stream specificato.</p> <p>Unità: millisecondi</p>
<code>GetMP4InitFragment.Requests</code>	<p>Il numero di richieste <code>GetMP4InitFragment</code> API per un determinato flusso.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>GetMP4InitFragment.Success</code>	<p>1 per ogni richiesta senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div data-bbox="748 352 1508 758"><p> Note</p><p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle AWS richieste, consulta Registrazione del riepilogo delle richieste e delle risposte.</p></div> <p>Unità: numero</p>
<code>GetMP4MediaFragment.Latency</code>	<p>La latenza delle chiamate <code>GetMP4MediaFragment</code> API per il nome dello stream specificato.</p> <p>Unità: millisecondi</p>
<code>GetMP4MediaFragment.Requests</code>	<p>Il numero di richieste <code>GetMP4MediaFragment</code> API per un determinato flusso.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>GetMP4MediaFragment.Success</code>	<p>1 per ogni richiesta senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div data-bbox="748 352 1507 758" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px;"><p> Note</p><p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle AWS richieste, consulta Registrazione del riepilogo delle richieste e delle risposte.</p></div> <p>Unità: numero</p>
<code>GetMP4MediaFragment.OutgoingBytes</code>	<p>Il numero totale di byte inviati dal servizio come parte dell'<code>GetMP4MediaFragment</code> API per un determinato flusso.</p> <p>Unità: byte</p>
<code>GetTSFragment.Latency</code>	<p>La latenza delle chiamate <code>GetTSFragment</code> API per il nome dello stream specificato.</p> <p>Unità: millisecondi</p>
<code>GetTSFragment.Requests</code>	<p>Il numero di richieste <code>GetTSFragment</code> API per un determinato flusso.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>GetTSFfragment.Success</code>	<p>1 per ogni richiesta senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div style="border: 1px solid #00a0e3; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p> Note</p> <p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle AWS richieste, consulta Registrazione del riepilogo delle richieste e delle risposte.</p> </div> <p>Unità: numero</p>
<code>GetTSFfragment.OutgoingBytes</code>	<p>Il numero totale di byte inviati dal servizio come parte dell'<code>GetTSFfragment</code> API per un determinato flusso.</p> <p>Unità: byte</p>
<code>GetDASHStreamingSessionURL.Latency</code>	<p>La latenza delle chiamate <code>GetDASHStreamingSessionURL</code> API per il nome dello stream specificato.</p> <p>Unità: millisecondi</p>
<code>GetDASHStreamingSessionURL.Requests</code>	<p>Il numero di richieste <code>GetDASHStreamingSessionURL</code> API per un determinato flusso.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>GetDASHStreamingSessionURL.Success</code>	<p>1 per ogni richiesta senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div data-bbox="748 352 1507 758" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px;"><p> Note</p><p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle AWS richieste, consulta Registrazione del riepilogo delle richieste e delle risposte.</p></div> <p>Unità: numero</p>
<code>GetDASHManifest.Latency</code>	<p>La latenza delle chiamate <code>GetDASHManifest</code> API per il nome dello stream specificato.</p> <p>Unità: millisecondi</p>
<code>GetDASHManifest.Requests</code>	<p>Il numero di richieste <code>GetDASHManifest</code> API per un determinato flusso.</p> <p>Unità: numero</p>

Parametro	Descrizione
<code>GetDASHManifest.Success</code>	<p>1 per ogni richiesta senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div data-bbox="748 352 1508 758"><p> Note</p><p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle AWS richieste, consulta Registrazione del riepilogo delle richieste e delle risposte.</p></div> <p>Unità: numero</p>
<code>GetClip.Latency</code>	<p>La latenza delle chiamate GetClip API per il nome del flusso video specificato.</p> <p>Unità: millisecondi</p>
<code>GetClip.Requests</code>	<p>Il numero di richieste GetClip API per un determinato flusso video.</p> <p>Unità: numero</p>

Parametro	Descrizione
GetClip.Success	<p>1 per ogni richiesta senza errori; 0 per ogni errore. Il valore medio indica la percentuale di successo.</p> <div style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin: 10px 0;"> <p>Note</p> <p>Gli errori includono errori 400 (utente) e 500 (sistema). Per ulteriori informazioni sull'attivazione di un riepilogo delle richieste e delle risposte, inclusi gli ID delle AWS richieste, consulta Registrazione del riepilogo delle richieste e delle risposte.</p> </div> <p>Unità: numero</p>
GetClip.OutgoingBytes	<p>Il numero totale di byte inviati dal servizio come parte dell' GetClip API per un determinato flusso video.</p> <p>Unità: byte</p>

CloudWatch indicazioni sulle metriche

CloudWatch le metriche possono aiutare a trovare risposte alle seguenti domande:

Argomenti

- [I dati raggiungono il servizio Amazon Kinesis Video Streams?](#)
- [Perché i dati non vengono inseriti correttamente dal servizio Amazon Kinesis Video Streams?](#)
- [Perché i dati non possono essere letti dal servizio Amazon Kinesis Video Streams alla stessa velocità con cui vengono inviati dal produttore?](#)
- [Perché non c'è video nella console oppure perché il video viene riprodotto in ritardo?](#)
- [Con quanto ritardo vengono letti i dati in tempo reale e perché il client è in ritardo rispetto all'inizio del flusso?](#)
- [Il client sta leggendo i dati dal flusso video di Kinesis e a che velocità?](#)
- [Perché il client non è in grado di leggere i dati dal flusso video di Kinesis?](#)

I dati raggiungono il servizio Amazon Kinesis Video Streams?

Parametri pertinenti:

- `PutMedia.IncomingBytes`
- `PutMedia.IncomingFragments`
- `PutMedia.IncomingFrames`

Cosa fare:

- Se si verifica un calo di questi parametri, controlla se l'applicazione sta ancora inviando dati al servizio.
- Verificare la larghezza di banda della rete. Se è insufficiente, potrebbe rallentare la velocità con cui il servizio riceve i dati.

Perché i dati non vengono inseriti correttamente dal servizio Amazon Kinesis Video Streams?

Parametri pertinenti:

- `PutMedia.Requests`
- `PutMedia.ConnectionErrors`
- `PutMedia.Success`
- `PutMedia.ErrorAckCount`

Cosa fare:

- Se c'è un aumento `PutMedia.ConnectionErrors`, guarda la risposta HTTP e i codici di errore ricevuti dal client di produzione per vedere quali errori si verificano durante la creazione della connessione.
- Se c'è un calo `PutMedia.Success` o un aumento del numero di `PutMedia.ErrorAckCount`, controlla il codice di errore ack nelle risposte ack inviate dal servizio per capire perché l'inserimento dei dati non riesce. [Per ulteriori informazioni, consulta `.Values.AckErrorCode`](#)

Perché i dati non possono essere letti dal servizio Amazon Kinesis Video Streams alla stessa velocità con cui vengono inviati dal produttore?

Parametri pertinenti:

- `PutMedia.FragmentIngestionLatency`
- `PutMedia.IncomingBytes`

Cosa fare:

- Se si verifica un calo di questi parametri, controlla la larghezza di banda di rete delle tue connessioni. Una connessione con larghezza di banda ridotta potrebbe causare un trasferimento lento dei dati al servizio.

Perché non c'è video nella console oppure perché il video viene riprodotto in ritardo?

Parametri pertinenti:

- `PutMedia.FragmentIngestionLatency`
- `PutMedia.FragmentPersistLatency`
- `PutMedia.Success`
- `ListFragments.Latency`
- `PutMedia.IncomingFragments`

Cosa fare:

- Se c'è un aumento `PutMedia.FragmentIngestionLatency` o una diminuzione `PutMedia.IncomingFragments`, controlla la larghezza di banda della rete e se i dati vengono ancora inviati.
- Se c'è un drop in `PutMedia.Success`, controlla i codici di errore ack. Per ulteriori informazioni, consulta [AckErrorCode.Values](#).
- Se c'è un aumento `PutMedia.FragmentPersistLatency` o `ListFragments.Latency`, è molto probabile che tu stia riscontrando un problema di servizio. Se la condizione persiste per un periodo di tempo prolungato, contatta il tuo contatto del servizio clienti per verificare se c'è un problema con il tuo servizio.

Con quanto ritardo vengono letti i dati in tempo reale e perché il client è in ritardo rispetto all'inizio del flusso?

Parametri pertinenti:

- `GetMedia.MillisBehindNow`
- `GetMedia.ConnectionErrors`
- `GetMedia.Success`

Cosa fare:

- Se si verifica un aumento `GetMedia.ConnectionErrors`, il consumatore potrebbe rimanere indietro nella lettura dello streaming a causa dei frequenti tentativi di riconnettersi allo streaming. Controllare i codici di risposta/errore HTTP restituiti per la richiesta `GetMedia`.
- Se si verifica un calo `GetMedia.Success`, probabilmente è dovuto al fatto che il servizio non è in grado di inviare i dati al consumatore, con conseguente interruzione della connessione e riconnessione tra i consumatori, con conseguente ritardo del consumatore rispetto allo stream.
- Se c'è un aumento `GetMedia.MillisBehindNow`, controlla i tuoi limiti di larghezza di banda per vedere se stai ricevendo i dati a una velocità inferiore a causa della larghezza di banda inferiore.

Il client sta leggendo i dati dal flusso video di Kinesis e a che velocità?

Parametri pertinenti:

- `GetMedia.OutgoingBytes`
- `GetMedia.OutgoingFragments`
- `GetMedia.OutgoingFrames`
- `GetMediaForFragmentList.OutgoingBytes`
- `GetMediaForFragmentList.OutgoingFragments`
- `GetMediaForFragmentList.OutgoingFrames`

Cosa fare:

- Queste metriche indicano la velocità con cui vengono letti i dati archiviati e in tempo reale.

Perché il client non è in grado di leggere i dati dal flusso video di Kinesis?

Parametri pertinenti:

- `GetMedia.ConnectionErrors`
- `GetMedia.Success`
- `GetMediaForFragmentList.Success`
- `PutMedia.IncomingBytes`

Cosa fare:

- Se c'è un aumento `GetMedia.ConnectionErrors`, guarda la risposta HTTP e i codici di errore restituiti dalla `GetMedia` richiesta. Per ulteriori informazioni, consulta [AckErrorCode.Values](#).
- Se stai cercando di leggere i dati più recenti o in tempo reale, controlla `PutMedia.IncomingBytes` se ci sono dati in arrivo nello stream che il servizio invia ai consumatori.
- Se si verifica un calo `GetMedia.Success` o `GetMediaForFragmentList.Success`, probabilmente, è dovuto al fatto che il servizio non è in grado di inviare i dati al consumatore. Se la condizione persiste per un periodo di tempo prolungato, contatta il tuo contatto del servizio clienti per verificare se c'è un problema con il tuo servizio.

Monitoraggio dell'agente Amazon Kinesis Video Streams Edge con CloudWatch

Puoi monitorare Amazon Kinesis Video Streams Edge Agent utilizzando CloudWatch Amazon, che raccoglie ed elabora i dati grezzi in metriche leggibili quasi in tempo reale. Queste statistiche vengono registrate per un periodo di 15 mesi. Con queste informazioni storiche, puoi avere una prospettiva migliore sulle prestazioni della tua applicazione Web o del servizio Amazon Kinesis Video Streams Edge Agent.

Per visualizzare le metriche, procedi come segue:

1. Accedi AWS Management Console e apri la CloudWatch console all'[indirizzo https://console.aws.amazon.com/cloudwatch/](https://console.aws.amazon.com/cloudwatch/).
2. Nella barra di navigazione a sinistra, in Metriche, seleziona Tutte le metriche.

3. Scegli la scheda Sfoglia, quindi seleziona lo spazio dei nomi `EdgeRuntimeAgentpersonalizzato`.

Amazon Kinesis Video Streams Edge Agent pubblica le seguenti metriche nel namespace: `EdgeRuntimeAgent`

Dimensioni	Stato	Descrizione
Nome dello stream, <code>RecordJob</code>	In esecuzione	Viene pubblicato continuamente quando <code>RecordJob</code> è in esecuzione. Unità: nessuna. «1" viene pubblicato finché si <code>RecordJob</code> trova in questo stato.
	<code>FatalError</code>	Pubblica in caso di errore <code>RecordJob</code> irreversibile. Unità: nessuna. «1" viene pubblicato una sola volta, quando si verifica questo evento. <div style="border: 1px solid #007bff; border-radius: 10px; padding: 10px; background-color: #e6f2ff;"> <p> Note Per ulteriori informazioni, consulta i log.</p> </div>
	Completato	Pubblica quando un <code>RecordJob</code> è completato. Unità: nessuna. «1" viene pubblicato una volta, quando si verifica questo evento.
Nome dello stream, <code>UploadJob</code>	In esecuzione	Viene pubblicato continuamente quando <code>UploadJob</code> è in esecuzione. Unità: nessuna. «1" viene pubblicato finché si <code>UploadJob</code> trova in questo stato.
	<code>FatalError</code>	Pubblica gli errori <code>UploadJob</code> fatali. Unità: nessuna. «1" viene pubblicato una volta, quando si verifica questo evento.

Dimensioni	Stato	Descrizione
		<div data-bbox="613 254 1507 428" style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; background-color: #e6f2ff;"> <p> Note Per ulteriori informazioni, consulta i log.</p> </div>
Nome del flusso	PercentageSpaceUsed	<p>Questa è la percentuale utilizzata rispetto allo spazio totale allocato nelle configurazioni di Amazon Kinesis Video Streams Edge Agent per la registrazione di contenuti multimediali. Per ulteriori informazioni, consulta the section called “LocalSizeConfig”.</p> <p>Unità: percentuale (scala 0—1).</p>
Nome oggetto	Vivo	<p>Pubblica ogni minuto da Amazon Kinesis Video Streams Edge Agent, indipendentemente dalle configurazioni in esecuzione su di esso.</p> <p>Questo può essere usato per capire se l'agente Amazon Kinesis Video Streams Edge è attivo e pronto ad accettare configurazioni.</p> <p>Unità: nessuna. «1" viene pubblicato ogni minuto.</p>
	RecordJobs.HealthyJobCount	<p>Numero totale di processi di registrazione in esecuzione e pianificati su Amazon Kinesis Video Streams Edge Agent.</p> <p>Unità: conteggio.</p>
	UploadJobs.HealthyJobCount	<p>Numero totale di processi di caricamento in esecuzione e pianificati su Amazon Kinesis Video Streams Edge Agent.</p> <p>Unità: conteggio.</p>

Dimensioni	Stato	Descrizione
	RecordJobs.UnhealthyJobCount	Conteggio totale dei processi di registrazione attualmente non riusciti. Unità: conta.
	UploadJobs.UnhealthyJobCount	Conteggio totale dei processi di caricamento attualmente errati. Unità: conta.
	RecordJobs.RunningJobCount	Numero totale di processi record in esecuzione attiva. Unità: conta.
	UploadJobs.RunningJobCount	Numero totale di processi di caricamento in esecuzione attiva. Unità: conta.
	RecordJobs.EdgeConfigCount	Numero totale di configurazioni di record in corso su Amazon Kinesis Video Streams Edge Agent. Unità: conteggio.
	UploadJobs.EdgeConfigCount	Numero totale di configurazioni di caricamento in corso su Amazon Kinesis Video Streams Edge Agent. Unità: conteggio.

CloudWatch indicazioni sui parametri per Amazon Kinesis Video Streams Edge Agent

CloudWatch le metriche possono essere utili per trovare risposte alle seguenti domande:

Argomenti

- [Amazon Kinesis Video Streams Edge Agent dispone di spazio sufficiente per registrare?](#)
- [L'agente Amazon Kinesis Video Streams Edge è attivo?](#)

- [Ci sono lavori non salutari?](#)
- [Qualche lavoro richiede un intervento esterno?](#)

Amazon Kinesis Video Streams Edge Agent dispone di spazio sufficiente per registrare?

Metriche pertinenti: `PercentageSpaceUsed`

Azione: nessuna azione richiesta.

L'agente Amazon Kinesis Video Streams Edge è attivo?

Metriche pertinenti: `Alive`

Azione: Se in qualsiasi momento smetti di ricevere questa metrica, significa che Amazon Kinesis Video Streams Edge Agent ha riscontrato una o più delle seguenti condizioni:

- Un problema di runtime dell'applicazione: vincoli di memoria o di altre risorse, bug e così via
- Il AWS IoT dispositivo su cui l'agente è in esecuzione al momento dello spegnimento, si è bloccato o è terminato
- Il AWS IoT dispositivo non dispone di connettività di rete

Ci sono lavori non salutari?

Parametri pertinenti:

- `RecordJobs.UnhealthyJobCount`
- `UploadJobs.UnhealthyJobCount`

Azione: Ispeziona i log e cerca la metrica. `FatalError`

- Se la `FatalError` metrica è presente, si è verificato un errore irreversibile ed è necessario riavviare manualmente il processo. Ispeziona i log e risolvi il problema prima di `StartEdgeConfigurationUpdate` riavviare manualmente il lavoro.
- Se la `FatalError` metrica non è presente, si è verificato un errore transitorio (non fatale) e Amazon Kinesis Video Streams Edge Agent sta riprovando a eseguire il lavoro.

Note

Per fare in modo che l'agente ritenti un lavoro con errore irreversibile, usa [the section called "StartEdgeConfigurationUpdate"](#)

Qualche lavoro richiede un intervento esterno?

Parametri pertinenti:

- `PercentageSpaceUsed`— Se questo valore supera un determinato valore, il processo di registrazione viene messo in pausa e riprende solo quando lo spazio è disponibile (quando i file multimediali non sono più disponibili). È possibile inviare una configurazione aggiornata con una versione superiore `MaxLocalMediaSizeInMB` per aggiornare immediatamente il lavoro.
- `RecordJob.FatalError/UploadJob.FatalError`— Analizzate i log dell'agente e inviate nuovamente la configurazione per riprendere il lavoro.

Azione: Effettua una chiamata API con la configurazione per riavviare i lavori che presentano questo problema.

Registrazione delle chiamate API Amazon Kinesis Video Streams con AWS CloudTrail

Amazon Kinesis Video Streams AWS CloudTrail funziona con un servizio che fornisce una registrazione delle azioni intraprese da un utente, da un ruolo Servizio AWS o da un utente in Amazon Kinesis Video Streams. CloudTrail acquisisce tutte le chiamate API per Amazon Kinesis Video Streams come eventi. Le chiamate acquisite includono chiamate dalla console Amazon Kinesis Video Streams e chiamate in codice alle operazioni dell'API Amazon Kinesis Video Streams. Se crei un trail, puoi abilitare la distribuzione continua di CloudTrail eventi a un bucket Amazon S3, inclusi gli eventi per Amazon Kinesis Video Streams. Se non configuri un percorso, puoi comunque visualizzare gli eventi più recenti nella CloudTrail console in Cronologia eventi. Utilizzando le informazioni raccolte da CloudTrail, puoi determinare la richiesta effettuata ad Amazon Kinesis Video Streams, l'indirizzo IP da cui è stata effettuata la richiesta, chi ha effettuato la richiesta, quando è stata effettuata e dettagli aggiuntivi.

Per ulteriori informazioni CloudTrail, incluso come configurarlo e abilitarlo, consulta la Guida per [l'AWS CloudTrail utente](#).

Amazon Kinesis Video Streams e CloudTrail

CloudTrail è abilitato sul tuo AWS account al momento della creazione dell'account. Quando si verifica un'attività di evento supportata in Amazon Kinesis Video Streams, tale attività viene registrata CloudTrail in un evento AWS insieme ad altri eventi di servizio nella cronologia degli eventi. Puoi visualizzare, cercare e scaricare eventi recenti nel tuo AWS account. Per ulteriori informazioni, consulta [Visualizzazione degli eventi con la cronologia degli CloudTrail eventi](#).

Per una registrazione continua degli eventi nel tuo AWS account, inclusi gli eventi per Amazon Kinesis Video Streams, crea un percorso. Un trail consente di CloudTrail inviare file di log a un bucket Amazon S3. Per impostazione predefinita, quando si crea un percorso nella console, questo sarà valido in tutte le Regioni AWS. Il trail registra gli eventi di tutte le regioni della AWS partizione e consegna i file di log al bucket Amazon S3 specificato. Inoltre, puoi configurarne altri Servizi AWS per analizzare ulteriormente e agire in base ai dati sugli eventi raccolti nei log. CloudTrail Per ulteriori informazioni, consulta gli argomenti seguenti:

- [Panoramica della creazione di un trail](#)
- [CloudTrail Servizi e integrazioni supportati](#)
- [Configurazione delle notifiche Amazon SNS per CloudTrail](#)
- [Ricezione di file di CloudTrail registro da più regioni](#) e [ricezione di file di CloudTrail registro da più account](#)

Amazon Kinesis Video Streams supporta la registrazione delle seguenti azioni come CloudTrail eventi nei file di registro:

- [CreateStream](#)
- [DeleteStream](#)
- [DescribeStream](#)
- [GetDataEndpoint](#)
- [ListStreams](#)
- [ListTagsForStream](#)
- [TagStream](#)
- [UntagStream](#)
- [UpdateDataRetention](#)
- [UpdateStream](#)

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 utente 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, vedete l'elemento [CloudTrailuserIdentity](#).

Esempio: voci dei file di registro di Amazon Kinesis Video Streams

Un trail è una configurazione che consente la distribuzione di eventi come file di log in un bucket Amazon S3 specificato dall'utente. CloudTrail i file di registro contengono una o più voci di registro. Un evento rappresenta una singola richiesta da un'fonte e include informazioni sull'azione richiesta, data e ora dell'azione, parametri richiesti e così via. CloudTrail i file di registro non sono una traccia ordinata delle chiamate API pubbliche, quindi non vengono visualizzati in un ordine specifico.

L'esempio seguente mostra una voce di CloudTrail registro che illustra l'[CreateStream](#)azione.

```
{
  "Records": [
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:user/Alice",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
      },
      "eventTime": "2018-05-25T00:16:31Z",
      "eventSource": "kinesisvideo.amazonaws.com",
      "eventName": "CreateStream",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "127.0.0.1",
      "userAgent": "aws-sdk-java/unknown-version Linux/x.xx",
      "requestParameters": {
        "streamName": "VideoStream",
        "dataRetentionInHours": 2,

```

```

        "mediaType": "mediaType",
        "kmsKeyId": "arn:aws:kms::us-east-1:123456789012:alias",
"deviceName": "my-device"
    },
    "responseElements": {
"streamARN":arn:aws:kinesisvideo:us-east-1:123456789012:stream/VideoStream/12345"
    },
    "requestID": "db6c59f8-c757-11e3-bc3b-57923b443c1c",
    "eventID": "b7acfd0-6ca9-4ee1-a3d7-c4e8d420d99b"
},
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:user/Alice",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2018-05-25:17:06Z",
    "eventSource": "kinesisvideo.amazonaws.com",
    "eventName": "DeleteStream",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "aws-sdk-java/unknown-version Linux/x.xx",
    "requestParameters": {
        "streamARN": "arn:aws:kinesisvideo:us-east-1:012345678910:stream/
VideoStream/12345",
        "currentVersion": "keqrjeqkj9"
    },
    "responseElements": null,
    "requestID": "f0944d86-c757-11e3-b4ae-25654b1d3136",
    "eventID": "0b2f1396-88af-4561-b16f-398f8eaea596"
},
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:user/Alice",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    }
}

```

```

    },
    "eventTime": "2014-04-19T00:15:02Z",
    "eventSource": " kinesisvideo.amazonaws.com",
    "eventName": "DescribeStream",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "aws-sdk-java/unknown-version Linux/x.xx",
    "requestParameters": {
      "streamName": "VideoStream"
    },
    "responseElements": null,
    "requestID": "a68541ca-c757-11e3-901b-cbcfe5b3677a",
    "eventID": "22a5fb8f-4e61-4bee-a8ad-3b72046b4c4d"
  },
  {
    "eventVersion": "1.05",
    "userIdentity": {
      "type": "IAMUser",
      "principalId": "EX_PRINCIPAL_ID",
      "arn": "arn:aws:iam::123456789012:user/Alice",
      "accountId": "123456789012",
      "accessKeyId": "EXAMPLE_KEY_ID",
      "userName": "Alice"
    },
    "eventTime": "2014-04-19T00:15:03Z",
    "eventSource": "kinesisvideo.amazonaws.com",
    "eventName": "GetDataEndpoint",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "aws-sdk-java/unknown-version Linux/x.xx",
    "requestParameters": {
      "streamName": "VideoStream",
      "aPName": "LIST_FRAGMENTS"
    },
    "responseElements": null,
    "requestID": "a6e6e9cd-c757-11e3-901b-cbcfe5b3677a",
    "eventID": "dcd2126f-c8d2-4186-b32a-192dd48d7e33"
  },
  {
    "eventVersion": "1.05",
    "userIdentity": {
      "type": "IAMUser",
      "principalId": "EX_PRINCIPAL_ID",

```

```
        "arn": "arn:aws:iam::123456789012:user/Alice",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2018-05-25T00:16:56Z",
    "eventSource": "kinesisvideo.amazonaws.com",
    "eventName": "ListStreams",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "aws-sdk-java/unknown-version Linux/x.xx",
    "requestParameters": {
        "maxResults": 100,
        "streamNameCondition": {"comparisonValue": "MyVideoStream"
comparisonOperator": "BEGINS_WITH"}}
    },
    "responseElements": null,
    "requestID": "e9f9c8eb-c757-11e3-bf1d-6948db3cd570",
    "eventID": "77cf0d06-ce90-42da-9576-71986fec411f"
}
]
}
```

Quote di servizio Kinesis Video Streams

Kinesis Video Streams ha le seguenti quote di servizio:

Important

Le seguenti quote di servizio sono o soft [s], che possono essere aggiornate inviando un ticket di supporto, o rigide [h], che non possono essere aumentate. Nelle tabelle seguenti verranno visualizzati [s] e [h] accanto alle quote di servizio individuali.

Quote di servizio API del piano di controllo

La sezione seguente descrive le quote di servizio per le API del piano di controllo. TPS è l'acronimo di Transactions per second.

Quando viene raggiunto un limite di richieste a livello di account o di risorsa, viene generato un `ClientLimitExceededException`

Quote dei servizi API del piano di controllo

API	Limite dell'account: Richiesta	Limite dell'account: Streams	Limite a livello di flusso	Eccezioni e note pertinenti
the section called "CreateStream"	50 TPS	10000 stream per account nelle regioni Stati Uniti orientali (Virginia settentrionale) e Stati Uniti occidentali (Oregon). 5000 stream per account	N/D	Questa API può essere invocata da dispositivi, CLI, accessi basati su SDK e dalla console. Se il flusso non è già esistente, va a buon fine una sola chiamata API.

API	Limite dell'account: Richiesta	Limite dell'account: Streams	Limite a livello di flusso	Eccezioni e note pertinenti
		<p>in tutte le altre regioni supportate.</p> <div data-bbox="591 474 792 1850" style="border: 1px solid #0070C0; border-radius: 10px; padding: 10px;"> <p> Note</p> <p>Questo limite può essere aumentato fino a 100.000 (o più) flussi per account. Accedi alla AWS Management Console all'indirizzo https://console.aws.amazon.com/ e invia un</p> </div>		

API	Limite dell'account: Richiesta	Limite dell'account: Streams	Limite a livello di flusso	Eccezioni e note pertinenti
		<p>caso di aumento dei limiti di servizio per Kinesis Video Streams per richiedere un aumento di questo limite.</p>		
the section called "DeleteEdgeConfiguration"	10 TPS [r]	N/D	1 TPS [h]	
the section called "DeleteStream"	50 TPS [r]	N/D	5 TPS [r]	

API	Limite dell'account: Richiesta	Limite dell'account: Streams	Limite a livello di flusso	Eccezioni e note pertinenti
the section called “Describe EdgeConfiguration”	50 TPS [r]	N/D	5 TPS [r]	
the section called “Describe ImageGenerationConfiguration”	50 TPS [r]	N/D	5 TPS [r]	
the section called “Describe MappedResourceConfiguration”	50 TPS [r]	N/D	5 TPS [r]	
the section called “Describe NotificationConfiguration”	50 TPS [r]	N/D	5 TPS [r]	
the section called “Describe Stream”	300 TPS [r]	N/D	5 TPS [r]	

API	Limite dell'account: Richiesta	Limite dell'account: Streams	Limite a livello di flusso	Eccezioni e note pertinenti
the section called "GetDataEndpoint"	300 TPS [r]	N/D	5 TPS [r]	Viene chiamata ogni 45 minuti per aggiornare il token di streaming per la maggior parte dei casi d'uso PutMedia/GetMedia. Il caching dei dati dell'endpoint è un'operazione sicura se l'applicazione ricarica i dati in caso di errore.
the section called "ListEdgeAgentConfigurations"	50 TPS [r]	N/D	N/D	
the section called "ListStreams"	50 TPS [r]	N/D	N/D	
the section called "ListTagsForStream"	50 TPS [r]	N/D	5 TPS [r]	
the section called "StartEdgeConfigurationUpdate"	10 TPS [r]	N/D	1 TPS [h]	
the section called "TagStream"	50 TPS [r]	N/D	5 TPS [r]	

API	Limite dell'account: Richiesta	Limite dell'account: Streams	Limite a livello di flusso	Eccezioni e note pertinenti
the section called “UntagStream”	50 TPS [r]	N/D	5 TPS [r]	
the section called “UpdateDataRetention”	50 TPS [r]	N/D	5 TPS [r]	
the section called “UpdateImageGenerationConfiguration”	50 TPS [r]	N/D	5 TPS [r]	
the section called “UpdateNotificationConfiguration”	50 TPS [r]	N/D	5 TPS [r]	
the section called “UpdateStream”	50 TPS [r]	N/D	5 TPS [r]	

Quote dei servizi API multimediali e archiviati

La sezione seguente descrive le quote di servizio per le API multimediali e archiviate.

Quando viene raggiunto un limite di richieste a livello di account o di risorsa, viene generato un `ClientLimitExceededException`

Quando viene raggiunto un limite a livello di connessione, viene generata un'eccezione `ConnectionLimitExceededException`.

Quando viene raggiunto un limite a livello di frammento, vengono generati i seguenti errori o messaggi ACK:

- Se un frammento è inferiore alla durata minima, viene restituito il messaggio ACK `MIN_FRAGMENT_DURATION_REACHED`.
- Se un frammento è superiore alla durata massima, viene restituito il messaggio ACK `MAX_FRAGMENT_DURATION_REACHED`.
- Se un frammento supera le dimensioni massime dei dati, viene restituito il messaggio ACK `MAX_FRAGMENT_SIZE`.
- Viene generata un'eccezione `FragmentLimitExceeded` se viene raggiunto un limite a livello di frammento in un'operazione `GetMediaForFragmentList`.

Quote dei servizi API Data Plane

API	Limite a livello di flusso	Limite a livello di connessione	Limite larghezza di banda	Limite a livello di frammento	Eccezioni e note pertinenti
the section called "PutMedia"	5 TPS [r]	1 [f]	12,5 MB/secondo o 100 Mbps [s] per stream	<ul style="list-style-type: none"> • Durata minima frammento : 1 secondo [r] • Durata massima del frammento : 20 	<p>Una tipica richiesta <code>PutMedia</code> conterrà i dati per alcuni secondi, generando un TPS minore per flusso.</p> <p>Se sono presenti più connessioni simultanee che superano le quote, viene accettata l'ultima connessione.</p>

API	Limite a livello di flusso	Limite a livello di connessione	Limite larghezza di banda	Limite a livello di frammento	Eccezioni e note pertinenti
				secondi [h] <ul style="list-style-type: none"> • Dimensione massima frammento : 50 MB [r] • Numero massimo di brani: 3 [s] • Numero massimo di frammenti inviati al secondo: 5 [h] • Limite massimo di metadati per frammenti : 10 tag [h] 	

API	Limite a livello di flusso	Limite a livello di connessione	Limite larghezza di banda	Limite a livello di frammento	Eccezioni e note pertinenti
the section called “GetClip”	N/D	N/D	Limite di dimensione 100 MB [h]	Numero massimo di frammenti: 200 [h]	
the section called “GetDASHStreamingSessionURL”	25 TPS [h]	N/D	N/D	N/D	
the section called “GetHLSStreamingSessionURL”	25 TPS [h]	N/D	N/D	N/D	
the section called “GetImages”	N/D	N/D	100 MB [h]	N/D	Il numero massimo di immagini per richiesta è 100 [h].

 **Note**

Il valore minimo per `SamplingInterval` è 200 millisecondi (ms), ovvero 5 immagini al secondo.

API	Limite a livello di flusso	Limite a livello di connessione	Limite larghezza di banda	Limite a livello di frammento	Eccezioni e note pertinenti
the section called "GetMedia"	5 TPS [r]	3 [f]	25 MB/sec. o 200 Mbps [f]	Numero massimo di frammenti inviati al secondo: 6 [h]	<p>Un client utente unico non dovrebbe aver bisogno di più di due o tre TPS perché dopo aver stabilito la connessione, l'applicazione dovrebbe leggere continuamente.</p> <p>Se un frammento tipico è di circa 5 MB, questo limite equivale a ~75 MBps per flusso video Kinesis. Un tale flusso avrebbe un bitrate in uscita doppio rispetto al massimo bitrate in entrata per i flussi.</p> <div style="border: 1px solid #0070C0; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p> Note</p> <p>GetMedia non viene utilizzato per la riproduzione HLS/DASH.</p> </div>

API	Limite a livello di flusso	Limite a livello di connessione	Limite larghezza di banda	Limite a livello di frammento	Eccezioni e note pertinenti
the section called "GetMediaForFragmentList"	N/D	5 [s]	25 MB/sec. o 200 Mbps [s]	Numero massimo di frammenti: 1.000 [h]	È possibile richiamare contemporaneamente cinque applicazioni che utilizzano frammenti. GetMediaForFragmentList Eventuali altre connessioni vengono rifiutate.

Quote dei servizi API del protocollo di riproduzione video

API	Limite a livello di sessione	Limite a livello di frammento
GetDash ManifestPlaylist	5 TPS [r]	Numero massimo di frammenti per playlist: 5000 [h]
GetHLS MasterPlaylist	5 TPS [r]	N/D
GetHLS MediaPlaylist	5 TPS [r]	Numero massimo di frammenti per playlist: 5000 [h]
Ottieni MP4 InitFragment	5 TPS [r]	N/D
Ottieni MP4 MediaFragment	20 TPS [h]	N/D
GetTSFragment	20 TPS [h]	N/D

Framment-metadati e framment-media quote

[Le API Kinesis Video Streams per l'accesso ai](#) contenuti multimediali archiviati vengono limitate in base al numero di frammenti richiesti anziché al numero di chiamate API. Le API sono limitate in base alla velocità sia dal numero di metadati dei frammenti che dal numero di frammenti multimediali

richiesti. I metadati dei frammenti e le quote multimediali dei frammenti vengono applicati per stream. In altre parole, le richieste di metadati o contenuti multimediali di frammento in uno stream non si applicano alle quote di un altro stream. Tuttavia, all'interno di un determinato flusso, ogni quota è condivisa tra più API. Ciò significa che, per un determinato flusso, le richieste di frammenti tra API diverse utilizzano la stessa quota. Quando viene superata la quota dei metadati dei frammenti o dei frammenti multimediali per uno stream, l'API restituisce un `ClientLimitExceededException`. Le tabelle seguenti mostrano come le API utilizzano ciascuno dei due tipi di quota. Per la seconda colonna di queste tabelle, supponiamo che se uno stream ha una quota di N, ciò significa che le API hanno N punti da utilizzare da quel tipo di quota per quel flusso. L'GetClipAPI appare in entrambe le tabelle.

Consumo di quote di metadati frammentati

API	Numero di punti di quota consumati per richiesta	Quota condivisa (N)
the section called “ListFragments”	Valore del <code>MaxResults</code> parametro	10000 punti di quota al secondo, per stream [h]
the section called “GetClip”	Numero di frammenti nella clip risultante	
GetHLSMediaPlaylist	Valore del parametro <code>MaxMediaPlaylistFragmentResults</code>	
GetDASHManifest	Valore del <code>MaxManifestFragmentResults</code> parametro	
the section called “GetImages”	Valore di oltre 400 (numero massimo di immagini richieste)	

Consumo di quote multimediali frammentate

API	Numero di punti di quota consumati per richiesta	Quota condivisa (N)
the section called “GetMediaForFragmentList”	Numero di frammenti nel parametro Fragments	500 punti di quota al secondo, per stream [h]
the section called “GetClip”	Numero di frammenti nella clip risultante	
GetMP4MediaFragment	1	
GetTSFragment	1	
the section called “GetImages”	Numero massimo di immagini richieste	

Ad esempio, con una quota di 500 frammenti multimediali al secondo, sono supportati i seguenti pattern di chiamata per un determinato flusso:

- 5 richieste al secondo GetClip con 100 frammenti in ogni clip.
- 100 richieste al secondo GetClip con 5 frammenti in ogni clip.
- 2 richieste al secondo per GetClip 100 frammenti in ogni clip e 3 richieste al secondo per GetMediaForFragmentList ogni clip.
- 400 richieste al secondo a GetMP4MediaFragment e 100 richieste al secondo a GetTSFragment.

Queste quote hanno un'importante implicazione per quanto riguarda il numero di sessioni HLS e MPEG-DASH che possono essere supportate per stream. Non c'è limite al numero di sessioni HLS e DASH che possono essere utilizzate dai lettori multimediali in un determinato momento. Pertanto, è importante che l'applicazione di riproduzione non consenta di utilizzare troppe sessioni contemporaneamente. I due esempi seguenti descrivono come determinare il numero di sessioni di riproduzione simultanee che possono essere supportate:

Esempio 1: streaming live

In uno scenario di live streaming con HLS con frammenti della durata di 1 secondo, una traccia audio e video e `MaxMediaPlaylistFragmentResults` impostato su cinque, un lettore multimediale effettua in genere due chiamate al `GetHLSMediaPlaylist` secondo. Una chiamata è per i metadati video più recenti e un'altra per i metadati audio corrispondenti. Le due chiamate utilizzano ciascuna cinque punti di quota di metadati frammentari. Inoltre, effettua due chiamate al secondo: una `GetMP4MediaFragment` per il video più recente e l'altra per l'audio corrispondente. Ogni chiamata utilizza un token multimediale a frammento singolo, quindi vengono consumati due token in totale.

In questo scenario, possono essere supportate fino a 250 sessioni di riproduzione simultanee. Con 250 sessioni, questo scenario consuma 2.500 punti di quota di metadati di frammenti al secondo (ben al di sotto della quota di 10.000) e 500 punti di quota multimediale di frammenti al secondo.

Esempio 2: riproduzione su richiesta

In uno scenario di riproduzione su richiesta di un evento passato con MPEG-DASH, una traccia audio e video `MaxManifestFragmentResults` impostata su 1.000, un lettore multimediale in genere chiama una `GetDASHManifest` volta all'inizio della sessione (consumando 1.000 punti di quota di metadati dei frammenti) e chiama `GetMP4MediaFragment` a una velocità massima di 5 volte al secondo (consumando 5 punti di quota multimediale per frammenti) fino al caricamento di tutti i frammenti. In questo scenario, è possibile avviare fino a 10 nuove sessioni al secondo (proprio alla quota di 10.000 frammenti di metadati al secondo) e fino a 100 sessioni possono caricare attivamente file multimediali frammentati a una velocità di 5 al secondo (proprio alla quota di 500 frammenti di file multimediali al secondo).

È possibile utilizzare `ArchivedFragmentsConsumed.Metadata` e `ArchivedFragmentsConsumed.Media` monitorare rispettivamente il consumo di metadati di frammenti e punti di quota dei frammenti multimediali. Per informazioni sul monitoraggio, vedere.

[Monitoraggio](#)

Quote di metadati per frammenti

Le seguenti quote di servizio si applicano all'aggiunta di metadati di frammenti ai frammenti in un flusso video Kinesis:

- Puoi anteporre a un frammento fino a 10 elementi metadati.
- La lunghezza del nome dei metadati del frammento può essere al massimo di 128 byte.
- La lunghezza del valore dei metadati del frammento può essere al massimo di 256 byte.

- Il nome di metadati di un frammento non può iniziare con la stringa "». AWS Se un elemento metadati viene aggiunto, il metodo `putFragmentMetadata` nel PIC restituisce un errore `STATUS_INVALID_METADATA_NAME` (codice errore `0x52000077`). La tua applicazione può ignorare l'errore (il PIC non aggiunge l'elemento metadati) oppure correggere l'errore.

Tag di streaming

Queste coppie chiave-valore dei metadati si applicano all'intera risorsa Kinesis Video Streams, non ai singoli frammenti contenuti in un flusso video Kinesis.

Ogni stream video Kinesis supporta fino a 50 tag.

Vedi [the section called "TagStream"](#) le restrizioni sulle chiavi e sui valori dei tag stream.

Risoluzione dei problemi relativi a Kinesis Video Streams

Utilizza le seguenti informazioni per risolvere i problemi più comuni riscontrati con Amazon Kinesis Video Streams.

Argomenti

- [Problemi generali](#)
- [Problemi relativi alle API](#)
- [Problemi HLS](#)
- [Problemi con Java](#)
- [Problemi con Producer Library](#)
- [Problemi relativi alla libreria Stream Parser](#)
- [Problemi di rete](#)

Problemi generali

Questa sezione descrive i problemi generali che potresti riscontrare quando lavori con Kinesis Video Streams.

Problemi

- [Latenza troppo elevata](#)

Latenza troppo elevata

La latenza potrebbe essere causata dalla durata dei frammenti inviati al servizio Kinesis Video Streams. Uno dei modi per ridurre la latenza tra producer e servizio è configurare la pipeline dei file multimediali per ottenere frammenti di più breve durata.

Per ridurre il numero di frame inviati in ogni frammento, riduci il seguente valore in:

kinesis_video_gstreamer_sample_app.cpp

```
g_object_set(G_OBJECT (data.encoder), "bframes", 0, "key-int-max", 45, "bitrate", 512,
NULL);
```

Note

Le latenze sono più elevate nel browser Mozilla Firefox a causa dell'implementazione interna del rendering video.

Problemi relativi alle API

Questa sezione descrive i problemi relativi alle API che potresti riscontrare quando lavori con Kinesis Video Streams.

Problemi

- [Errore: "Opzioni sconosciute"](#)
- [Errore: "Unable to determine service/operation name to be authorized" \(Impossibile determinare il nome del servizio/operazione da autorizzare\)](#)
- [Errore: "Failed to put a frame in the stream" \(Impossibile inserire un fotogramma nel flusso\)](#)
- [Errore: «Il servizio ha chiuso la connessione prima della ricezione del messaggio finale AckEvent »](#)
- [Errore "STATUS_STORE_OUT_OF_MEMORY"](#)

Errore: "Opzioni sconosciute"

Se le richieste `GetMedia` e `GetMediaForFragmentList` non riescono, può essere generato questo errore:

```
Unknown options: <filename>.mkv
```

Questo errore si verifica se l'hai configurato AWS CLI con un output tipo di `json`. Riconfigurare il AWS CLI con il tipo di output predefinito (`none`). Per informazioni sulla configurazione di AWS CLI, consulta [configure](#) nel AWS CLI Command Reference.

Errore: "Unable to determine service/operation name to be authorized" (Impossibile determinare il nome del servizio/operazione da autorizzare)

Se la richiesta `GetMedia` non riesce, può essere generato questo errore:

```
Unable to determine service/operation name to be authorized
```

Questo errore può verificarsi se l'endpoint non è specificato correttamente. Quando ricevi l'endpoint, assicurati di includere il seguente parametro nella `GetDataEndpoint` chiamata, a seconda dell'API da chiamare:

```
--api-name GET_MEDIA  
--api-name PUT_MEDIA  
--api-name GET_MEDIA_FOR_FRAGMENT_LIST  
--api-name LIST_FRAGMENTS
```

Errore: "Failed to put a frame in the stream" (Impossibile inserire un fotogramma nel flusso)

Se la richiesta `PutMedia` non riesce, può essere generato questo errore:

```
Failed to put a frame in the stream
```

Questo errore può verificarsi se la connettività o le autorizzazioni non sono disponibili per il servizio. Esegui quanto segue in e verifica che le informazioni sullo stream possano essere recuperate: AWS CLI

```
aws kinesisvideo describe-stream --stream-name StreamName --endpoint https://  
ServiceEndpoint.kinesisvideo.region.amazonaws.com
```

Se la chiamata fallisce, consulta [Risoluzione dei problemi AWS CLI degli errori](#) per ulteriori informazioni.

Errore: «Il servizio ha chiuso la connessione prima della ricezione del messaggio finale `AckEvent` »

Se la richiesta `PutMedia` non riesce, può essere generato questo errore:

```
com.amazonaws.SdkClientException: Service closed connection before final AckEvent was  
received
```

Questo errore può verificarsi se l'elemento `PushbackInputStream` non è implementato correttamente. Verifica che i `unread()` metodi siano implementati correttamente.

Errore "STATUS_STORE_OUT_OF_MEMORY"

Se la richiesta PutMedia non riesce, può essere generato questo errore:

```
The content store is out of memory.
```

Questo errore si verifica quando allo store dei contenuti non è allocato spazio a sufficienza. Per ovviare, aumentare il valore di `StorageInfo.storageSize`. Per ulteriori informazioni, consulta [StorageInfo](#).

Problemi HLS

Se lo streaming video non viene riprodotto correttamente, consulta [the section called "Risoluzione dei problemi relativi a HLS"](#).

Problemi con Java

Questa sezione descrive come risolvere i problemi Java più comuni riscontrati durante l'utilizzo di Kinesis Video Streams.

Problemi

- [Abilitazione dei log di Java](#)

Abilitazione dei log di Java

Per risolvere i problemi relativi agli esempi e alle librerie Java, è utile abilitare ed esaminare i log di debug. Per abilitarli, procedere come segue:

1. Aggiungere `log4j` al file `pom.xml` nel nodo `dependencies`:

```
<dependency>
  <groupId>log4j</groupId>
  <artifactId>log4j</artifactId>
  <version>1.2.17</version>
</dependency>
```

2. Nella directory `target/classes`, creare un file denominato `log4j.properties` con il seguente contenuto:

```
# Root logger option
log4j.rootLogger=DEBUG, stdout

# Redirect log messages to console
log4j.appender.stdout=org.apache.log4j.ConsoleAppender
log4j.appender.stdout.Target=System.out
log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
log4j.appender.stdout.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:
%L - %m%n

log4j.logger.org.apache.http.wire=DEBUG
```

I log di debug vengono quindi stampati nella console IDE.

Problemi con Producer Library

Questa sezione descrive i problemi che possono verificarsi durante l'utilizzo della [Producer Library](#).

Problemi

- [Impossibile compilare l'SDK producer](#)
- [Il flusso video non viene visualizzato nella console](#)
- [Errore "Security token included in the request is invalid" durante lo streaming dei dati con l'applicazione demo GStreamer](#)
- [Errore "Failed to submit frame to Kinesis Video client"](#)
- [L'applicazione GStreamer viene arrestata con il messaggio "streaming stopped, reason not negotiated" su OS X](#)
- [Errore "Failed to allocate heap" durante la creazione del client Kinesis Video nella demo GStreamer su Raspberry Pi](#)
- [Errore "Illegal Instruction" durante l'esecuzione della demo GStreamer su Raspberry Pi](#)
- [La videocamera non viene caricata su Raspberry Pi](#)
- [Impossibile trovare la videocamera su macOS High Sierra](#)
- [File jni.h non trovato durante la compilazione su macOS High Sierra](#)
- [Errori di curl durante l'esecuzione dell'applicazione demo GStreamer](#)
- [Asserzione timestamp/intervallo in fase di runtime su Raspberry Pi](#)

- [Asserzione su `gst_value_set_fraction_range_full` su Raspberry Pi](#)
- [STATUS_MKV_INVALID_ANNEXB_NALU_IN_FRAME_DATA \(0x3200000d\) error on Android](#)
- [È stata raggiunta la durata massima del frammento \(errore\)](#)
- [Errore "Invalid thing name passed \(Passato nome di cosa non valido\)" quando si utilizza l'autorizzazione IoT](#)

Impossibile compilare l'SDK producer

Controllare che le librerie richieste siano presenti nel percorso. Per eseguire una verifica, utilizzare il comando seguente:

```
env | grep LD_LIBRARY_PATH
LD_LIBRARY_PATH=/home/local/awslabs/amazon-kinesis-video-streams-producer-sdk-cpp/
kinesis-video-native-build/downloads/local/lib
```

Il flusso video non viene visualizzato nella console

Perché sia visualizzato nella console, il flusso video deve essere codificato con H.264 nel formato AvCC. Se il flusso non viene visualizzato, verificare quanto segue:

- I [Flag di adattamento del NAL](#) sono impostati su `NAL_ADAPTATION_ANNEXB_NALS` | `NAL_ADAPTATION_ANNEXB_CPD_NALS` se il flusso originale è in formato Annex-B. Questo è il valore di default nel costruttore `StreamDefinition`.
- I dati privati del codec sono forniti correttamente. Per H.264, si tratta del set di parametri di sequenza (SPS) e del set di parametri delle immagini (PPS). A seconda dell'origine del contenuto multimediale, questi dati potrebbero essere recuperati dall'origine separatamente o codificati nel fotogramma.

Molti flussi elementari sono nel formato seguente, dove `Ab` è il codice di inizio (001 o 0001) Annex-B:

```
Ab(Sps)Ab(Pps)Ab(I-frame)Ab(P/B-frame) Ab(P/B-frame)... Ab(Sps)Ab(Pps)Ab(I-frame)Ab(P/
B-frame) Ab(P/B-frame)
```

Il CPD (Codec Private Data), se H.264 è presente nello stream come SPS e PPS, può essere adattato al formato AVCC. A meno che la pipeline multimediale non fornisca il CPD separatamente, l'applicazione può estrarre il CPD dal frame cercando il primo frame Idr (che dovrebbe contenere

SPS e PPS), estrarre i due NALU (che saranno Ab (Sps) Ab (Pps)) e impostarli nel CPD `inStreamDefinition`.

Errore "Security token included in the request is invalid" durante lo streaming dei dati con l'applicazione demo GStreamer

Questo errore indica che è presente un problema con le credenziali. Verificare quanto segue:

- Se si utilizzano credenziali di sicurezza temporanee, è necessario specificare il token di sessione.
- Verificare che le credenziali temporanee non siano scadute.
- Verificare che siano impostati gli opportuni diritti.
- Su macOS, verificare che non vi siano credenziali memorizzate nella cache in Keychain.

Errore "Failed to submit frame to Kinesis Video client"

Questo errore indica che i timestamp non sono impostati in modo corretto nel flusso sorgente.

Eeguire quanto segue:

- Utilizzare l'SDK di esempio più recente, che potrebbe includere un aggiornamento per la risoluzione del problema.
- Imposta lo stream di alta qualità su un bitrate più alto e correggi eventuali jitter nel flusso sorgente, se la fotocamera lo supporta.

L'applicazione GStreamer viene arrestata con il messaggio "streaming stopped, reason not-negotiated" su OS X

Lo streaming potrebbe essere arrestato su OS X con il seguente messaggio:

```
Debugging information: gstbasesrc.c(2939): void gst_base_src_loop(GstPad *) (): /
GstPipeline:test-pipeline/GstAutoVideoSrc:source/GstAVFVideoSrc:source-actual-src-
avfvide:
streaming stopped, reason not-negotiated (-4)
```

Una possibile soluzione alternativa consiste nel rimuovere i parametri del frame rate dalla `gst_caps_new_simple` chiamata in: `kinesis_video_gstreamer_sample_app.cpp`

```
GstCaps *h264_caps = gst_caps_new_simple("video/x-h264",
                                         "profile", G_TYPE_STRING, "baseline",
                                         "stream-format", G_TYPE_STRING, "avc",
                                         "alignment", G_TYPE_STRING, "au",
                                         "width", GST_TYPE_INT_RANGE, 320, 1920,
                                         "height", GST_TYPE_INT_RANGE, 240, 1080,
                                         "framerate", GST_TYPE_FRACTION_RANGE, 0,
                                         1, 30, 1,
                                         NULL);
```

Errore "Failed to allocate heap" durante la creazione del client Kinesis Video nella demo GStreamer su Raspberry Pi

L'applicazione di esempio GStreamer cerca di allocare 512 MB di RAM, che potrebbero non essere disponibili nel sistema. È possibile diminuire tale allocazione riducendo il seguente valore in `KinesisVideoProducer.cpp`:

```
device_info.storageInfo.storageSize = 512 * 1024 * 1024;
```

Errore "Illegal Instruction" durante l'esecuzione della demo GStreamer su Raspberry Pi

Se riscontri il seguente errore durante l'esecuzione della demo di GStreamer, verifica di aver compilato l'applicazione per la versione corretta del tuo dispositivo. (Ad esempio, verifica che non stai compilando per Raspberry Pi 3 quando utilizzi Raspberry Pi 2.)

```
INFO - Initializing curl.
Illegal instruction
```

La videocamera non viene caricata su Raspberry Pi

Per controllare se la videocamera viene caricata, eseguire quanto segue:

```
ls /dev/video*
```

Se l'operazione non ha esito, eseguire quanto segue:

```
vcgencmd get_camera
```

L'output visualizzato dovrebbe essere simile al seguente:

```
supported=1 detected=1
```

Se il driver non rileva la videocamera, procedere come segue:

1. Controllare la configurazione fisica della videocamera e verificare che sia collegata correttamente.
2. Eseguire quanto segue per aggiornare il firmware:

```
sudo rpi-update
```

3. Riavviare il dispositivo.
4. Eseguire quanto segue per caricare il driver:

```
sudo modprobe bcm2835-v4l2
```

5. Verificare che la videocamera sia stata rilevata:

```
ls /dev/video*
```

Impossibile trovare la videocamera su macOS High Sierra

Su macOS High Sierra, l'applicazione demo non è in grado di trovare la videocamera se ne è disponibile più di una.

File jni.h non trovato durante la compilazione su macOS High Sierra

Per risolvere questo errore, aggiornare l'installazione di Xcode alla versione più recente.

Errori di curl durante l'esecuzione dell'applicazione demo GStreamer

Per risolvere gli errori di curl durante l'esecuzione dell'applicazione demo GStreamer, copiare [questo file di certificato](#) in `/etc/ssl/cert.pem`.

Assertione timestamp/intervallo in fase di runtime su Raspberry Pi

Se si verifica un'asserzione di intervallo timestamp in fase di runtime, aggiornare il firmware e riavviare il dispositivo:

```
sudo rpi-update
$ sudo reboot
```

Assertione su `gst_value_set_fraction_range_full` su Raspberry Pi

La seguente assertione viene visualizzata se è in esecuzione il servizio `uv4l`:

```
gst_util_fraction_compare (numerator_start, denominator_start, numerator_end,
denominator_end) < 0' failed
```

In questo caso, interrompere il servizio `uv4l` e riavviare l'applicazione.

STATUS_MKV_INVALID_ANNEXB_NALU_IN_FRAME_DATA (0x3200000d) error on Android

Il seguente errore viene visualizzato se [Flag di adattamento del NAL](#) non sono corretti per il flusso di file multimediali:

```
putKinesisVideoFrame(): Failed to put a frame with status code 0x3200000d
```

Se si verifica questo errore, specificare il flag `.withNalAdaptationFlags` corretto per i file multimediali (ad esempio, `NAL_ADAPTATION_ANNEXB_CPD_NALS`). Fornire questo flag nella seguente riga di [Android Producer Library](#):

<https://github.com/aws-labs/aws-sdk-android-samples/blob/master/src/main/java/com/amazonaws/kinesisvideo/demoapp/fragment/.java#L169> [AmazonKinesisVideoDemoAppStreamConfigurationFragment](#)

È stata raggiunta la durata massima del frammento (errore)

Questo errore si verifica quando un frammenti di file multimediale in un flusso supera il limite massimo di durata. Vedi il limite massimo di durata del frammento nella [the section called "Quote dei servizi API multimediali e archiviati"](#) sezione.

Per risolvere il problema, prova a eseguire queste operazioni:

- Se stai usando una webcam/fotocamera USB, procedi in uno dei seguenti modi:
 - Se utilizzi la frammentazione basata su frame chiave, imposta l'encoder in modo che fornisca i fotogrammi chiave entro 10 secondi.

- Se non utilizzi la frammentazione basata su frame chiave, quando definisci lo stream in [Fase 2: Scrivere ed esaminare il codice](#), imposta il limite massimo di durata del frammento su un valore inferiore a 10 secondi.
- Se utilizzi codificatori software (come x264) nella pipeline GStreamer, puoi impostare l'attributo su un valore entro 10 secondi. key-int-max Ad esempio, impostate su key-int-max 60, con fps impostati su 30, per abilitare i keyframe ogni 2 secondi.
- Se utilizzate una fotocamera RPI, impostate l'attributo keyframe-interval su un valore inferiore a 10 secondi.
- Se utilizzi una telecamera IP (RTSP), imposta la dimensione GOP su 60.

Errore "Invalid thing name passed (Passato nome di cosa non valido)" quando si utilizza l'autorizzazione IoT

Per evitare questo errore (HTTP Error 403: Response: {"message":"Invalid thing name passed"}) quando utilizzi le credenziali IoT per l'autorizzazione, assicurati che il valore di stream-name (un parametro obbligatorio dell'kvssinkelemento) sia identico al valore di iot-thingname. Per ulteriori informazioni, consulta [Riferimento ai parametri dell'elemento GStreamer](#).

Problemi relativi alla libreria Stream Parser

Questa sezione descrive i problemi che possono verificarsi durante l'utilizzo della [Libreria parser del flusso](#).

Problemi

- [Impossibile accedere a un singolo fotogramma del flusso](#)
- [Errore di decodifica frammento](#)

Impossibile accedere a un singolo fotogramma del flusso

Per accedere a un singolo frame da una sorgente di streaming nella tua applicazione consumer, verifica che lo stream contenga i dati privati del codec corretti. Per informazioni sul formato dei dati in un flusso, consultare [Modello di dati](#).

[Per informazioni su come utilizzare i dati privati del codec per accedere a un frame, consultate il seguente file di test sul GitHub sito Web: .java KinesisVideoRendererExampleTest](#)

Errore di decodifica frammento

Se i frammenti non sono correttamente codificati in un formato H.264 e in un livello supportato dal browser, durante la riproduzione del flusso nella console può essere generato il seguente errore:

```
Fragment Decoding Error
There was an error decoding the video data. Verify that the stream contains valid H.264
content
```

In questo caso, verificare quanto segue:

- La risoluzione dei fotogrammi corrisponde alla risoluzione specificata nei dati privati del codec (Codec Private Data).
- Il profilo e il livello H.264 dei fotogrammi codificati corrispondono al profilo e al livello specificati nei dati privati del codec (Codec Private Data).
- Il browser supporta la combinazione profilo/livello. La maggior parte dei browser attuali supporta tutte le combinazioni di profilo e livello.
- I timestamp sono accurati e nell'ordine corretto, senza creazione di duplicati in corso.
- L'applicazione codifica i dati dei fotogrammi utilizzando il formato H.264.

Problemi di rete

Se si riscontrano errori di connessione, come «Timeout della connessione» o «Connessione non riuscita», quando si tenta di connettersi a Kinesis Video Streams, ciò potrebbe essere dovuto alle restrizioni dell'intervallo di indirizzi IP nella configurazione di rete.

[Se la configurazione prevede restrizioni sull'intervallo di indirizzi IP per Kinesis Video Streams, aggiorna la configurazione di rete per consentire gli intervalli di indirizzi IP di Kinesis Video Streams.](#)

[Per ulteriori informazioni, consulta Intervalli IP.AWS](#) Per ricevere una notifica quando gli intervalli IP cambiano, segui la [procedura di sottoscrizione](#).

Cronologia documento per Amazon Kinesis Video Streams

La tabella seguente descrive le modifiche importanti alla documentazione dall'ultima versione di Amazon Kinesis Video Streams.

- Versione API più recente: 29-11-2017
- Ultimo aggiornamento della documentazione: 27 giugno 2023

Modifica	Descrizione	Data
Connessione edge-to-cloud di Amazon Kinesis Video Streams Edge Agent	Rilascio di una nuova funzionalità. Per ulteriori informazioni, consulta Agente Edge .	27 giugno 2023
Guida introduttiva: invio di dati a un flusso video Kinesis	Tutorial di base per l'invio di dati multimediali da una videocamera a un flusso video Kinesis. Per ulteriori informazioni, consulta Inviare dati a un flusso video di Amazon Kinesis .	21 gennaio 2019
Streaming di metadati	Puoi utilizzare Producer SDK per incorporare i metadati in un flusso video Kinesis. Per ulteriori informazioni, consulta Utilizzo dei metadati di streaming con Kinesis Video Streams .	28 settembre 2018
Logging di C++ Producer SDK	Puoi configurare il logging per le applicazioni C++ Producer SDK. Per ulteriori informazioni, consulta Utilizzo	18 luglio 2018

Modifica	Descrizione	Data
	della registrazione con C++ Producer SDK.	
Streaming di video HLS	Ora puoi visualizzare uno streaming video Kinesis utilizzando HTTP Live Streaming. Per ulteriori informazioni, consulta Riproduzione di Kinesis Video Streams.	13 luglio 2018
Streaming da un'origine RTSP	Applicazione di esempio per Kinesis Video Streams che viene eseguita in un contenitore Docker e trasmette video da una sorgente RTSP. Per ulteriori informazioni, consulta RTSP e Docker.	20 giugno 2018
Plugin C++ Producer SDK GStreamer	Mostra come creare un C++ Producer Library da utilizzare come destinazione GStreamer . Per ulteriori informazioni, consulta Plugin GStreamer - kvssink.	15 giugno 2018
Documentazione di riferimento sui callback dell'SDK producer	Documentazione di riferimento per i callback utilizzati da Librerie per produttori di Kinesis Video Streams. Per ulteriori informazioni, consulta Richiamate SDK di Producer.	12 giugno 2018

Modifica	Descrizione	Data
Requisiti di sistema	Documentazione relativa ai requisiti di memoria e di storage per i dispositivi producer e SDK. Per ulteriori informazioni, consulta Requisiti di sistema di Kinesis Video Streams .	30 maggio 2018
CloudTrail supporto	Documentazione CloudTrail da utilizzare per monitorare e l'utilizzo delle API. Per ulteriori informazioni, consulta Registrazione delle chiamate API Amazon Kinesis Video Streams con AWS CloudTrail .	24 maggio 2018
Documentazione di riferimento sulle strutture SDK producer	Documentazione di riferimento per le strutture utilizzate da Librerie per produttori di Kinesis Video Streams . Per ulteriori informazioni, consulta Strutture SDK di Producer e Strutture dei flussi video Kinesis .	7 maggio 2018
Documentazione di esempio di Renderer	Documentazione per l'applicazione di esempio Renderer, che mostra come decodificare e visualizzare i frame da un flusso video Kinesis. Per ulteriori informazioni, consulta Esempio: analisi e rendering di frammenti di Kinesis Video Streams .	15 marzo 2018

Modifica	Descrizione	Data
Documentazione di riferimento sui limiti dell'SDK producer	Le informazioni sui limiti per le operazioni in C++ Producer Library . Per ulteriori informazioni, consulta Limiti di Producer SDK .	13 marzo 2018
Monitoraggio	Informazioni sul monitoraggio delle metriche e delle chiamate API di Kinesis Video Streams tramite Amazon e. CloudWatch AWS CloudTrail Per ulteriori informazioni, consulta Monitoraggio dei flussi video di Amazon Kinesis .	5 febbraio 2018
Documentazione di riferimento per i flag di adattamento del Network Abstraction Layer (NAL)	Informazioni sull'impostazione dei flag di adattamento NAL nell'utilizzo di streaming di video. Per ulteriori informazioni, consulta Flag di adattamento del NAL .	15 gennaio 2018
Supporto Android per lo streaming di video	Kinesis Video Streams ora supporta lo streaming di video da dispositivi Android. Per ulteriori informazioni, consulta Android Producer Library .	12 gennaio 2018

Modifica	Descrizione	Data
Documentazione di esempio di Kinesis Video	Documentazione per l'applicazione di esempio Kinesis Video, che mostra come utilizzarla Libreria di analisi dei flussi video Kinesis in un'applicazione. Per ulteriori informazioni, consulta KinesisVideoExample .	9 gennaio 2018
Rilasciata la documentazione di Kinesis Video Streams	Rilascio iniziale della Developer Guide di Amazon Kinesis Video Streams.	29 novembre 2017

Documentazione di riferimento delle API

Le sezioni sotto questo nodo contengono la documentazione di riferimento dell'API. Utilizza il sommario nel riquadro a sinistra per accedere alle diverse sezioni di riferimento dell'API.

Operazioni

Le seguenti azioni sono supportate da Amazon Kinesis Video Streams:

- [CreateSignalingChannel](#)
- [CreateStream](#)
- [DeleteEdgeConfiguration](#)
- [DeleteSignalingChannel](#)
- [DeleteStream](#)
- [DescribeEdgeConfiguration](#)
- [DescribeImageGenerationConfiguration](#)
- [DescribeMappedResourceConfiguration](#)
- [DescribeMediaStorageConfiguration](#)
- [DescribeNotificationConfiguration](#)
- [DescribeSignalingChannel](#)
- [DescribeStream](#)
- [GetDataEndpoint](#)
- [GetSignalingChannelEndpoint](#)
- [ListEdgeAgentConfigurations](#)
- [ListSignalingChannels](#)
- [ListStreams](#)
- [ListTagsForResource](#)
- [ListTagsForStream](#)
- [StartEdgeConfigurationUpdate](#)
- [TagResource](#)
- [TagStream](#)
- [UntagResource](#)

- [UntagStream](#)
- [UpdateDataRetention](#)
- [UpdateImageGenerationConfiguration](#)
- [UpdateMediaStorageConfiguration](#)
- [UpdateNotificationConfiguration](#)
- [UpdateSignalingChannel](#)
- [UpdateStream](#)

Le seguenti azioni sono supportate da Amazon Kinesis Video Streams Media:

- [GetMedia](#)
- [PutMedia](#)

Le seguenti azioni sono supportate da Amazon Kinesis Video Streams Archived Media:

- [GetClip](#)
- [GetDASHStreamingSessionURL](#)
- [GetHLSStreamingSessionURL](#)
- [GetImages](#)
- [GetMediaForFragmentList](#)
- [ListFragments](#)

Le seguenti azioni sono supportate da Amazon Kinesis Video Signaling Channels:

- [GetIceServerConfig](#)
- [SendAlexaOfferToMaster](#)

Le seguenti azioni sono supportate da Amazon Kinesis Video WebRTC Storage:

- [JoinStorageSession](#)

Flusso di video Amazon Kinesis

Le seguenti azioni sono supportate da Amazon Kinesis Video Streams:

- [CreateSignalingChannel](#)
- [CreateStream](#)
- [DeleteEdgeConfiguration](#)
- [DeleteSignalingChannel](#)
- [DeleteStream](#)
- [DescribeEdgeConfiguration](#)
- [DescribeImageGenerationConfiguration](#)
- [DescribeMappedResourceConfiguration](#)
- [DescribeMediaStorageConfiguration](#)
- [DescribeNotificationConfiguration](#)
- [DescribeSignalingChannel](#)
- [DescribeStream](#)
- [GetDataEndpoint](#)
- [GetSignalingChannelEndpoint](#)
- [ListEdgeAgentConfigurations](#)
- [ListSignalingChannels](#)
- [ListStreams](#)
- [ListTagsForResource](#)
- [ListTagsForStream](#)
- [StartEdgeConfigurationUpdate](#)
- [TagResource](#)
- [TagStream](#)
- [UntagResource](#)
- [UntagStream](#)
- [UpdateDataRetention](#)
- [UpdateImageGenerationConfiguration](#)
- [UpdateMediaStorageConfiguration](#)
- [UpdateNotificationConfiguration](#)
- [UpdateSignalingChannel](#)
- [UpdateStream](#)

CreateSignalingChannel

Servizio: Amazon Kinesis Video Streams

Crea un canale di segnalazione.

CreateSignalingChannel è un'operazione asincrona.

Sintassi della richiesta

```
POST /createSignalingChannel HTTP/1.1
Content-type: application/json

{
  "ChannelName": "string",
  "ChannelType": "string",
  "SingleMasterConfiguration": {
    "MessageTtlSeconds": number
  },
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ChannelName

Un nome per il canale di segnalazione che stai creando. Deve essere unico per ogni mano Account AWS . Regione AWS

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: sì

[ChannelType](#)

Un tipo di canale di segnalazione che stai creando. Attualmente, SINGLE_MASTER è l'unico tipo di canale supportato.

▪Tipo: stringa

Valori validi: SINGLE_MASTER | FULL_MESH

Campo obbligatorio: no

[SingleMasterConfiguration](#)

Una struttura contenente la configurazione per il tipo di SINGLE_MASTER canale.

Tipo: oggetto [SingleMasterConfiguration](#)

Campo obbligatorio: no

[Tags](#)

Un insieme di tag (coppie chiave-valore) da associare a questo canale.

Tipo: matrice di oggetti [Tag](#)

Membri dell'array: numero minimo di 0 elementi. Numero massimo di 50 item.

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "ChannelARN": "string"
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

ChannelARN

L'Amazon Resource Name (ARN) del canale creato.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non disponi delle autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

AccountChannelLimitExceededException

Hai raggiunto il limite massimo di canali di segnalazione attivi per questo scopo Account AWS in questa regione.

Codice di stato HTTP: 400

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceInUseException

Quando l'ingresso `StreamARN` o `ChannelARN` l'ingresso `CLOUD_STORAGE_MODE` sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'`StreamARN` ingresso fornito `ChannelARN` o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'`DescribeMediaStorageConfigurationAPI` per determinare a cosa è mappato lo stream di un determinato canale.
2. L'`DescribeMappedResourceConfigurationAPI` per determinare il canale su cui è mappato lo stream specificato.
3. L'`DescribeStream` o l'`DescribeSignalingChannelAPI` per determinare lo stato della risorsa.

Codice di stato HTTP: 400

TagsPerResourceExceededLimitException

Hai superato il limite di tag che puoi associare alla risorsa. Uno stream video Kinesis può supportare fino a 50 tag.

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

CreateStream

Servizio: Amazon Kinesis Video Streams

Crea un nuovo flusso video Kinesis.

Quando crei un nuovo flusso, Kinesis Video Streams assegna al flusso un numero di versione.

Quando modifichi i metadati del flusso, Kinesis Video Streams aggiorna la versione.

CreateStream è un'operazione asincrona.

Per ulteriori informazioni sul funzionamento del servizio, consulta [Come funziona](#).

È necessario avere le autorizzazioni per l'operazione `KinesisVideo:CreateStream`.

Sintassi della richiesta

```
POST /createStream HTTP/1.1
Content-type: application/json

{
  "DataRetentionInHours": number,
  "DeviceName": "string",
  "KmsKeyId": "string",
  "MediaType": "string",
  "StreamName": "string",
  "Tags": {
    "string" : "string"
  }
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

[DataRetentionInHours](#)

Il numero di ore in cui desideri conservare i dati nello stream. Kinesis Video Streams mantiene i dati in un archivio dati associato al flusso.

Il valore predefinito è 0, che indica che il flusso non conserva i dati.

Quando il `DataRetentionInHours` valore è 0, i consumatori possono comunque consumare i frammenti che rimangono nel buffer dell'host del servizio, che ha un limite di tempo di conservazione di 5 minuti e un limite di memoria di conservazione di 200 MB. I frammenti vengono rimossi dal buffer quando viene raggiunto uno dei due limiti.

Tipo: integer

Intervallo valido: valore minimo di 0.

Campo obbligatorio: no

DeviceName

Il nome del dispositivo che sta scrivendo nello stream.

Note

Nell'implementazione corrente, Kinesis Video Streams non utilizza questo nome.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

KmsKeyId

L'ID della chiave AWS Key Management Service (AWS KMS) che desideri che Kinesis Video Streams utilizzi per crittografare i dati di streaming.

Se non viene specificato alcun ID di chiave, viene utilizzata la chiave predefinita, gestita da Kinesis Video () `AWS/kinesisvideo`.

Per ulteriori informazioni, consulta. [DescribeKey](#)

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 1. La lunghezza massima è 2048 caratteri.

Modello: . +

Campo obbligatorio: no

MediaType

Il tipo di file multimediale dello stream. Gli utenti dello stream possono utilizzare queste informazioni durante l'elaborazione dello stream. Per ulteriori informazioni sui tipi di file multimediali, consulta [Tipi di file multimediali](#). Se scegli di specificare il MediaType, consulta [Requisiti di denominazione per le linee guida](#).

I valori validi di esempio includono «video/h264» e «video/h264, audio/aac».

Questo parametro è facoltativo; il valore predefinito è (o vuoto in JSON). null

■Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: `[\w\-\.\+]+/[\w\-\.\+]+(,[\w\-\.\+]+/[\w\-\.\+]+)*`

Campo obbligatorio: no

StreamName

Un nome per lo stream che stai creando.

Il nome dello stream è un identificatore dello stream e deve essere univoco per ogni account e regione.

■Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: sì

Tags

Un elenco di tag da associare allo stream specificato. Ogni tag è una coppia chiave-valore (il valore è opzionale).

Tipo: mappatura stringa a stringa

Voci sulla mappa: numero massimo di 50 elementi.

Limitazioni di lunghezza della chiave: la lunghezza minima è 1. La lunghezza massima è 128 caratteri.

Modello di chiave: $^{\wedge}([\text{p}\{\text{L}\}\text{p}\{\text{Z}\}\text{p}\{\text{N}\}_\cdot : / = + \backslash - @]^*)^{\$}$

Limiti di lunghezza del valore: lunghezza minima di 0. La lunghezza massima è 256 caratteri.

Modello di valore: $[\text{p}\{\text{L}\}\text{p}\{\text{Z}\}\text{p}\{\text{N}\}_\cdot : / = + \backslash - @]^*$

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "StreamARN": "string"
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

StreamARN

Il nome della risorsa Amazon (ARN) del flusso.

- Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: $\text{arn}:[\text{a-z}\backslash\text{d}-]^+:\text{kinesisvideo}:[\text{a-z}\backslash\text{0-9}-]^+:[\text{0-9}]^+:[\text{a-z}]^+/\text{[\text{a-zA-Z}\backslash\text{0-9}_\cdot -]^+}/[\text{0-9}]^+$

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccountStreamLimitExceededException

Il numero di stream creati per l'account è troppo alto.

Codice di stato HTTP: 400

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

DeviceStreamLimitExceededException

Non ancora disponibile.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

InvalidDeviceException

Non ancora disponibile.

Codice di stato HTTP: 400

ResourceInUseException

Quando l'ingresso `StreamARN` o `ChannelARN` l'ingresso `CLOUD_STORAGE_MODE` sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'`StreamARN` ingresso fornito `ChannelARN` o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'`DescribeMediaStorageConfigurationAPI` per determinare a cosa è mappato lo stream di un determinato canale.
2. L'`DescribeMappedResourceConfigurationAPI` per determinare il canale su cui è mappato lo stream specificato.
3. L'`DescribeStream` o l'`DescribeSignalingChannelAPI` per determinare lo stato della risorsa.

Codice di stato HTTP: 400

TagsPerResourceExceededLimitException

Hai superato il limite di tag che puoi associare alla risorsa. Uno stream video Kinesis può supportare fino a 50 tag.

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

DeleteEdgeConfiguration

Servizio: Amazon Kinesis Video Streams

Un'API asincrona che elimina la configurazione edge esistente di uno stream, nonché i media corrispondenti dall'Edge Agent.

Quando richiami questa API, lo stato di sincronizzazione è impostato su DELETING. Viene avviato un processo di eliminazione, in cui i lavori edge attivi vengono interrotti e tutti i file multimediali vengono eliminati dal dispositivo periferico. Il tempo di eliminazione varia a seconda della quantità totale di file multimediali archiviati. Se il processo di eliminazione fallisce, lo stato di sincronizzazione cambia in DELETE_FAILED. Dovrai riprovare a eseguire l'eliminazione.

Quando il processo di eliminazione è completato con successo, la configurazione del bordo non è più accessibile.

Note

Questa API non è disponibile nella regione AWS Africa (Città del Capo), af-south-1.

Sintassi della richiesta

```
POST /deleteEdgeConfiguration HTTP/1.1
Content-type: application/json
```

```
{
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

StreamARN

Il nome della risorsa Amazon (ARN) del flusso. Specificare il o il StreamName StreamARN

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome del flusso da cui eliminare la configurazione del bordo. Specificate il `StreamName` o il `StreamARN`.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non si dispone delle autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

StreamEdgeConfigurationNotFoundException

L'eccezione viene renderizzata quando Amazon Kinesis Video Stream non riesce a trovare la configurazione edge dello stream specificata.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

DeleteSignalingChannel

Servizio: Amazon Kinesis Video Streams

Elimina un canale di segnalazione specificato. DeleteSignalingChannel è un'operazione asincrona. Se non si specifica la versione corrente del canale, viene eliminata la versione più recente.

Sintassi della richiesta

```
POST /deleteSignalingChannel HTTP/1.1
Content-type: application/json

{
  "ChannelARN": "string",
  "CurrentVersion": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ChannelARN

L'Amazon Resource Name (ARN) del canale di segnalazione che desideri eliminare.

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+

Campo obbligatorio: sì

CurrentVersion

La versione corrente del canale di segnalazione che desideri eliminare. È possibile ottenere la versione corrente richiamando le operazioni DescribeSignalingChannel o ListSignalingChannels API.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 64 caratteri.

Modello: [a-zA-Z0-9]+

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non disponete delle autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceInUseException

Quando l'ingresso `StreamARN` o `ChannelARN` l'ingresso `CLOUD_STORAGE_MODE` sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'`StreamARN` ingresso fornito `ChannelARN` o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'`DescribeMediaStorageConfigurationAPI` per determinare a cosa è mappato lo stream di un determinato canale.
2. L'`DescribeMappedResourceConfigurationAPI` per determinare il canale su cui è mappato lo stream specificato.
3. L'`DescribeStream` o l'`DescribeSignalingChannelAPI` per determinare lo stato della risorsa.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

VersionMismatchException

La versione dello stream che hai specificato non è la versione più recente. Per ottenere la versione più recente, usa l'[DescribeStreamAPI](#).

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

DeleteStream

Servizio: Amazon Kinesis Video Streams

Elimina un flusso video Kinesis e i dati in esso contenuti.

Questo metodo contrassegna lo stream per l'eliminazione e rende immediatamente inaccessibili i dati in esso contenuti.

Per assicurarti di disporre della versione più recente dello stream prima di eliminarlo, puoi specificare la versione dello stream. Kinesis Video Streams assegna una versione a ogni stream. Quando aggiorni uno stream, Kinesis Video Streams assegna un nuovo numero di versione. Per ottenere la versione di streaming più recente, utilizza l'API. `DescribeStream`

Questa operazione richiede l'autorizzazione per l'operazione `KinesisVideo:DeleteStream`.

Sintassi della richiesta

```
POST /deleteStream HTTP/1.1
Content-type: application/json

{
  "CurrentVersion": "string",
  "StreamARN": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

CurrentVersion

Facoltativo: la versione dello stream che desideri eliminare.

Specificate la versione come protezione per assicurarvi che stiate eliminando lo stream corretto. Per ottenere la versione in streaming, usa l'`DescribeStreamAPI`.

Se non specificato, `CreationTime` viene controllato solo il prima di eliminare lo stream.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 64 caratteri.

Modello: [a-zA-Z0-9]+

Campo obbligatorio: no

StreamARN

L'Amazon Resource Name (ARN) dello stream che desideri eliminare.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+

Campo obbligatorio: sì

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire questa operazione.

Codice di stato HTTP: 401

ResourceInUseException

Quando l'ingresso `StreamARN` o `ChannelARN` l'ingresso `CLOUD_STORAGE_MODE` sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'`StreamARN` ingresso fornito `ChannelARN` o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'`DescribeMediaStorageConfigurationAPI` per determinare a cosa è mappato lo stream di un determinato canale.
2. L'`DescribeMappedResourceConfigurationAPI` per determinare il canale su cui è mappato lo stream specificato.
3. L'`DescribeStream` o l'`DescribeSignalingChannelAPI` per determinare lo stato della risorsa.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

VersionMismatchException

La versione dello stream che hai specificato non è la versione più recente. Per ottenere la versione più recente, usa l'[DescribeStreamAPI](#).

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)

- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

DescribeEdgeConfiguration

Servizio: Amazon Kinesis Video Streams

Descrive la configurazione edge di uno stream impostata utilizzando l'`StartEdgeConfigurationUpdateAPI` e lo stato più recente dei processi di registrazione e upload dell'agente edge. Utilizza questa API per ottenere lo stato della configurazione e determinare se la configurazione è sincronizzata con Edge Agent. Utilizza questa API per valutare lo stato di salute dell'Edge Agent.

Note

Questa API non è disponibile nella regione AWS Africa (Città del Capo), `af-south-1`.

Sintassi della richiesta

```
POST /describeEdgeConfiguration HTTP/1.1
Content-type: application/json
```

```
{
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

StreamARN

Il nome della risorsa Amazon (ARN) del flusso. Specificare il o il `StreamName` `StreamARN`

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome dello stream di cui si desidera aggiornare la configurazione edge. Specificate il StreamName o ilStreamARN.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "CreationTime": number,
  "EdgeAgentStatus": {
    "LastRecorderStatus": {
      "JobStatusDetails": "string",
      "LastCollectedTime": number,
      "LastUpdatedTime": number,
      "RecorderStatus": "string"
    },
    "LastUploaderStatus": {
      "JobStatusDetails": "string",
      "LastCollectedTime": number,
      "LastUpdatedTime": number,
      "UploaderStatus": "string"
    }
  },
  "EdgeConfig": {
    "DeletionConfig": {
      "DeleteAfterUpload": boolean,
      "EdgeRetentionInHours": number,
      "LocalSizeConfig": {
        "MaxLocalMediaSizeInMB": number,
        "StrategyOnFullSize": "string"
      }
    }
  }
}
```

```

    },
    "HubDeviceArn": "string",
    "RecorderConfig": {
      "MediaSourceConfig": {
        "MediaUriSecretArn": "string",
        "MediaUriType": "string"
      },
      "ScheduleConfig": {
        "DurationInSeconds": number,
        "ScheduleExpression": "string"
      }
    },
    "UploaderConfig": {
      "ScheduleConfig": {
        "DurationInSeconds": number,
        "ScheduleExpression": "string"
      }
    }
  },
  "FailedStatusDetails": "string",
  "LastUpdatedTime": number,
  "StreamARN": "string",
  "StreamName": "string",
  "SyncStatus": "string"
}

```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

CreationTime

Il timestamp in cui è stata creata per la prima volta la configurazione edge di uno stream.

Tipo: Timestamp

EdgeAgentStatus

Un oggetto che contiene i dettagli più recenti sullo stato dei lavori di registrazione e caricamento di un agente edge. Utilizza queste informazioni per determinare lo stato attuale di un agente edge.

Tipo: oggetto [EdgeAgentStatus](#)

[EdgeConfig](#)

Una descrizione della configurazione edge dello stream che verrà utilizzata per la sincronizzazione con il componente Edge Agent IoT Greengrass. Il componente Edge Agent verrà eseguito su un dispositivo Hub IoT configurato presso la tua sede.

Tipo: oggetto [EdgeConfig](#)

[FailedStatusDetails](#)

Una descrizione dello stato di errore generato.

▪Tipo: stringa

[LastUpdatedTime](#)

Il timestamp in cui la configurazione edge di uno stream è stata aggiornata l'ultima volta.

Tipo: Timestamp

[StreamARN](#)

Il nome della risorsa Amazon (ARN) del flusso.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

[StreamName](#)

Il nome dello stream da cui è stata aggiornata la configurazione edge.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

[SyncStatus](#)

Lo stato più recente dell'aggiornamento della configurazione edge.

▪Tipo: stringa

Valori validi: SYNCING | ACKNOWLEDGED | IN_SYNC | SYNC_FAILED | DELETING | DELETE_FAILED | DELETING_ACKNOWLEDGED

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

StreamEdgeConfigurationNotFoundException

L'eccezione viene renderizzata quando Amazon Kinesis Video Stream non riesce a trovare la configurazione edge dello stream specificata.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

DescribeImageGenerationConfiguration

Servizio: Amazon Kinesis Video Streams

Ottiene il ImageGenerationConfiguration flusso video Kinesis per un determinato flusso video.

Sintassi della richiesta

```
POST /describeImageGenerationConfiguration HTTP/1.1
Content-type: application/json

{
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

[StreamARN](#)

L'Amazon Resource Name (ARN) del flusso video Kinesis da cui recuperare la configurazione di generazione dell'immagine. È necessario specificare il o il. StreamName StreamARN

-Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+

Campo obbligatorio: no

[StreamName](#)

Il nome dello stream da cui recuperare la configurazione di generazione dell'immagine. È necessario specificare il StreamName o ilStreamARN.

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "ImageGenerationConfiguration": {
    "DestinationConfig": {
      "DestinationRegion": "string",
      "Uri": "string"
    },
    "Format": "string",
    "FormatConfig": {
      "string": "string"
    },
    "HeightPixels": number,
    "ImageSelectorType": "string",
    "SamplingInterval": number,
    "Status": "string",
    "WidthPixels": number
  }
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

ImageGenerationConfiguration

La struttura che contiene le informazioni necessarie per la distribuzione delle immagini Kinesis Video Stream (KVS). Se questa struttura è nulla, la configurazione verrà eliminata dallo stream.

Tipo: oggetto ImageGenerationConfiguration

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)

- [AWS SDK per Ruby V3](#)

DescribeMappedResourceConfiguration

Servizio: Amazon Kinesis Video Streams

Restituisce le informazioni più aggiornate sullo stream. L'`streamName` o `streamARN` deve essere fornito nell'input.

Sintassi della richiesta

```
POST /describeMappedResourceConfiguration HTTP/1.1
Content-type: application/json
```

```
{
  "MaxResults": number,
  "NextToken": "string",
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

[MaxResults](#)

Il numero massimo di risultati da restituire nella risposta.

Tipo: integer

Intervallo valido: valore fisso pari a 1.

Campo obbligatorio: no

[NextToken](#)

Il token da fornire nella richiesta successiva, per ottenere un altro batch di risultati.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: `[a-zA-Z0-9+/=]*`

Campo obbligatorio: no

StreamARN

Il nome della risorsa Amazon (ARN) del flusso.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome del flusso .

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "MappedResourceConfigurationList": [
    {
      "ARN": "string",
      "Type": "string"
    }
  ],
  "NextToken": "string"
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

[MappedResourceConfigurationList](#)

Una struttura che incapsula o contiene le proprietà di configurazione dell'archiviazione multimediale.

Tipo: matrice di oggetti [MappedResourceConfigurationListItem](#)

Membri dell'array: numero minimo di 0 elementi. Numero massimo di 1 elemento.

[NextToken](#)

Il token utilizzato nella NextToken richiesta per recuperare il set successivo di risultati.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: [a-zA-Z0-9+/=]*

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

DescribeMediaStorageConfiguration

Servizio: Amazon Kinesis Video Streams

Restituisce le informazioni più aggiornate sul canale. Specificare ChannelName o ChannelARN nell'input.

Sintassi della richiesta

```
POST /describeMediaStorageConfiguration HTTP/1.1
Content-type: application/json
```

```
{
  "ChannelARN": "string",
  "ChannelName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ChannelARN

L'Amazon Resource Name (ARN) del canale.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+

Campo obbligatorio: no

ChannelName

Nome del canale.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "MediaStorageConfiguration": {
    "Status": "string",
    "StreamARN": "string"
  }
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

MediaStorageConfiguration

Una struttura che incapsula o contiene le proprietà di configurazione dello storage multimediale.

Tipo: oggetto [MediaStorageConfiguration](#)

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non disponete delle autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

DescribeNotificationConfiguration

Servizio: Amazon Kinesis Video Streams

Ottiene il NotificationConfiguration file per un determinato flusso video Kinesis.

Sintassi della richiesta

```
POST /describeNotificationConfiguration HTTP/1.1
Content-type: application/json

{
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

StreamARN

L'Amazon Resource Name (ARN) del flusso video Kinesis da cui desideri recuperare la configurazione delle notifiche. Devi specificare il o StreamArnStreamName.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+

Campo obbligatorio: no

StreamName

Il nome dello stream da cui recuperare la configurazione della notifica. È necessario specificare il StreamName o ilStreamARN.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "NotificationConfiguration": {
    "DestinationConfig": {
      "Uri": "string"
    },
    "Status": "string"
  }
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

NotificationConfiguration

La struttura che contiene le informazioni richieste per le notifiche. Se la struttura è nulla, la configurazione verrà eliminata dallo stream.

Tipo: oggetto [NotificationConfiguration](#)

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

DescribeSignalingChannel

Servizio: Amazon Kinesis Video Streams

Restituisce le informazioni più aggiornate sul canale di segnalazione. Devi specificare il nome o l'Amazon Resource Name (ARN) del canale che desideri descrivere.

Sintassi della richiesta

```
POST /describeSignalingChannel HTTP/1.1
Content-type: application/json
```

```
{
  "ChannelARN": "string",
  "ChannelName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ChannelARN

L'ARN del canale di segnalazione che si desidera descrivere.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

ChannelName

Il nome del canale di segnalazione che si desidera descrivere.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "ChannelInfo": {
    "ChannelARN": "string",
    "ChannelName": "string",
    "ChannelStatus": "string",
    "ChannelType": "string",
    "CreationTime": number,
    "SingleMasterConfiguration": {
      "MessageTtlSeconds": number
    },
    "Version": "string"
  }
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

ChannelInfo

Una struttura che incapsula i metadati e le proprietà del canale di segnalazione specificato.

Tipo: oggetto [ChannelInfo](#)

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

DescribeStream

Servizio: Amazon Kinesis Video Streams

Restituisce le informazioni più aggiornate sullo stream specificato. È necessario specificare il `StreamName` o il `StreamARN`.

Sintassi della richiesta

```
POST /describeStream HTTP/1.1
Content-type: application/json

{
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

StreamARN

Il nome della risorsa Amazon (ARN) del flusso.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome del flusso .

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "StreamInfo": {
    "CreationTime": number,
    "DataRetentionInHours": number,
    "DeviceName": "string",
    "KmsKeyId": "string",
    "MediaType": "string",
    "Status": "string",
    "StreamARN": "string",
    "StreamName": "string",
    "Version": "string"
  }
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

[StreamInfo](#)

Un oggetto che descrive lo stream.

Tipo: oggetto [StreamInfo](#)

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire questa operazione.

Codice di stato HTTP: 401

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

GetDataEndpoint

Servizio: Amazon Kinesis Video Streams

Ottiene un endpoint per uno stream specificato per la lettura o la scrittura. Utilizzate questo endpoint nell'applicazione per leggere dal flusso specificato (usando le `GetMediaForFragmentList` operazioni `GetMedia` o) o scrivervi (usando l'operazione `PutMedia`).

Note

All'endpoint restituito non è stato aggiunto il nome dell'API. Il client deve aggiungere il nome dell'API all'endpoint restituito.

Nella richiesta, specifica lo stream con `StreamName` o `StreamARN`.

Sintassi della richiesta

```
POST /getDataEndpoint HTTP/1.1
Content-type: application/json
```

```
{
  "APIName": "string",
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

APIName

Il nome dell'azione API per la quale ottenere un endpoint.

▪Tipo: stringa

Valori validi: PUT_MEDIA | GET_MEDIA | LIST_FRAGMENTS |
GET_MEDIA_FOR_FRAGMENT_LIST | GET_HLS_STREAMING_SESSION_URL |
GET_DASH_STREAMING_SESSION_URL | GET_CLIP | GET_IMAGES

Campo obbligatorio: sì

StreamARN

L'Amazon Resource Name (ARN) dello stream per cui desideri ottenere l'endpoint. Devi specificare questo parametro o un StreamName nella richiesta.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+

Campo obbligatorio: no

StreamName

Il nome dello stream per il quale vuoi ottenere l'endpoint. È necessario specificare questo parametro o un StreamARN nella richiesta.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "DataEndpoint": "string"
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

DataEndpoint

Il valore dell'endpoint. Per leggere o scrivere dati dallo stream, specifica questo endpoint nell'applicazione.

▪Tipo: stringa

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire questa operazione.

Codice di stato HTTP: 401

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

GetSignalingChannelEndpoint

Servizio: Amazon Kinesis Video Streams

Fornisce un endpoint per il canale di segnalazione specificato per inviare e ricevere messaggi. Questa API utilizza il parametro `SingleMasterChannelEndpointConfiguration` di input, che consiste nelle proprietà `Protocols` e `Role`.

`Protocols` viene utilizzato per determinare il meccanismo di comunicazione. Ad esempio, se si specifica `WSS` come protocollo, questa API produce un endpoint websocket sicuro. Se si specifica `HTTPS` come protocollo, questa API genera un endpoint `HTTPS`. Se lo specifichi `WEBRTC` come protocollo, ma il canale di segnalazione non è configurato per l'ingestione, riceverai l'errore.

`InvalidArgumentException`

`Role` determina le autorizzazioni di messaggistica. Un `MASTER` ruolo fa sì che questa API generi un endpoint che un client può utilizzare per comunicare con qualsiasi spettatore del canale. Un `VIEWER` ruolo fa sì che questa API generi un endpoint che un client può utilizzare per comunicare solo con un `MASTER`.

Sintassi della richiesta

```
POST /getSignalingChannelEndpoint HTTP/1.1
Content-type: application/json

{
  "ChannelARN": "string",
  "SingleMasterChannelEndpointConfiguration": {
    "Protocols": [ "string " ],
    "Role": "string"
  }
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ChannelARN

L'Amazon Resource Name (ARN) del canale di segnalazione per il quale desideri ottenere un endpoint.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: sì

SingleMasterChannelEndpointConfiguration

Una struttura contenente la configurazione dell'endpoint per il tipo di canale. SINGLE_MASTER

Tipo: oggetto [SingleMasterChannelEndpointConfiguration](#)

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "ResourceEndpointList": [
    {
      "Protocol": "string",
      "ResourceEndpoint": "string"
    }
  ]
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

[ResourceEndpointList](#)

Un elenco di endpoint per il canale di segnalazione specificato.

Tipo: matrice di oggetti [ResourceEndpointListItem](#)

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceInUseException

Quando l'ingresso `StreamARN` o `ChannelARN` l'ingresso `CLOUD_STORAGE_MODE` sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'`StreamARN` ingresso fornito `ChannelARN` o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'`DescribeMediaStorageConfigurationAPI` per determinare a cosa è mappato lo stream di un determinato canale.
2. L'`DescribeMappedResourceConfigurationAPI` per determinare il canale su cui è mappato lo stream specificato.
3. L'`DescribeStream` o l'`DescribeSignalingChannelAPI` per determinare lo stato della risorsa.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

ListEdgeAgentConfigurations

Servizio: Amazon Kinesis Video Streams

Restituisce una matrice di configurazioni edge associate all'Edge Agent specificato.

Nella richiesta, è necessario specificare l'Edge Agent. HubDeviceArn

Note

Questa API non è disponibile nella regione AWS Africa (Città del Capo), af-south-1.

Sintassi della richiesta

```
POST /listEdgeAgentConfigurations HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "HubDeviceArn": "string",
  "MaxResults": number,
  "NextToken": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

HubDeviceArn

L'oggetto «Internet of Things (IoT)» Arn dell'agente edge.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:iot:[a-z0-9-]+:[0-9]+:thing/[a-zA-Z0-9_.-]+`

Campo obbligatorio: sì

MaxResults

Il numero massimo di configurazioni edge da restituire nella risposta. Il predefinito è 5.

Tipo: integer

Intervallo valido: valore minimo di 1. valore massimo pari a 10.

Campo obbligatorio: no

NextToken

Se si specifica questo parametro, quando il risultato di un'ListEdgeAgentConfigurationsoperazione viene troncato, la chiamata restituisce il NextToken valore nella risposta. Per ottenere un altro batch di configurazioni edge, fornisci questo token nella tua prossima richiesta.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: [a-zA-Z0-9+/=]*

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "EdgeConfigs": [
    {
      "CreationTime": number,
      "EdgeConfig": {
        "DeletionConfig": {
          "DeleteAfterUpload": boolean,
          "EdgeRetentionInHours": number,
          "LocalSizeConfig": {
            "MaxLocalMediaSizeInMB": number,
            "StrategyOnFullSize": "string"
          }
        }
      },
      "HubDeviceArn": "string",
```

```

    "RecorderConfig": {
      "MediaSourceConfig": {
        "MediaUriSecretArn": "string",
        "MediaUriType": "string"
      },
      "ScheduleConfig": {
        "DurationInSeconds": number,
        "ScheduleExpression": "string"
      }
    },
    "UploaderConfig": {
      "ScheduleConfig": {
        "DurationInSeconds": number,
        "ScheduleExpression": "string"
      }
    }
  },
  "FailedStatusDetails": "string",
  "LastUpdatedTime": number,
  "StreamARN": "string",
  "StreamName": "string",
  "SyncStatus": "string"
}
],
"NextToken": "string"
}

```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

[EdgeConfigs](#)

Una descrizione della configurazione edge di un singolo flusso.

Tipo: matrice di oggetti [ListEdgeAgentConfigurationsEdgeConfig](#)

[NextToken](#)

Se la risposta viene troncata, la chiamata restituisce questo elemento con un determinato token. Per ottenere il prossimo batch di configurazioni edge, usa questo token nella tua prossima richiesta.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: [a-zA-Z0-9+/=]*

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire questa operazione.

Codice di stato HTTP: 401

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)

- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

ListSignalingChannels

Servizio: Amazon Kinesis Video Streams

Restituisce una serie di `ChannelInfo` oggetti. Ogni oggetto descrive un canale di segnalazione. Per recuperare solo i canali che soddisfano una condizione specifica, è possibile specificare a `ChannelNameCondition`

Sintassi della richiesta

```
POST /listSignalingChannels HTTP/1.1
Content-type: application/json

{
  "ChannelNameCondition": {
    "ComparisonOperator": "string",
    "ComparisonValue": "string"
  },
  "MaxResults": number,
  "NextToken": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

[ChannelNameCondition](#)

Facoltativo: restituisce solo i canali che soddisfano una condizione specifica.

Tipo: oggetto [ChannelNameCondition](#)

Campo obbligatorio: no

[MaxResults](#)

Il numero massimo di canali da restituire nella risposta. L'impostazione predefinita è 500.

Tipo: integer

Intervallo valido: valore minimo di 1. Valore massimo pari a 10000.

Campo obbligatorio: no

NextToken

Se si specifica questo parametro, quando il risultato di un'ListSignalingChannelsoperazione viene troncato, la chiamata restituisce il valore NextToken nella risposta. Per ottenere un altro batch di canali, fornisci questo token nella tua prossima richiesta.

-Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: [a-zA-Z0-9+/=]*

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "ChannelInfoList": [
    {
      "ChannelARN": "string",
      "ChannelName": "string",
      "ChannelStatus": "string",
      "ChannelType": "string",
      "CreationTime": number,
      "SingleMasterConfiguration": {
        "MessageTtlSeconds": number
      },
      "Version": "string"
    }
  ],
  "NextToken": "string"
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

[ChannelInfoList](#)

Un array di oggetti `ChannelInfo`.

Tipo: matrice di oggetti [ChannelInfo](#)

[NextToken](#)

Se la risposta viene troncata, la chiamata restituisce questo elemento con un token. Per ottenere il prossimo batch di stream, usa questo token nella tua prossima richiesta.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: `[a-zA-Z0-9+/=]*`

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

ListStreams

Servizio: Amazon Kinesis Video Streams

Restituisce una serie di `StreamInfo` oggetti. Ogni oggetto descrive un flusso. Per recuperare solo i flussi che soddisfano una condizione specifica, puoi specificare a. `StreamNameCondition`

Sintassi della richiesta

```
POST /listStreams HTTP/1.1
Content-type: application/json

{
  "MaxResults": number,
  "NextToken": "string",
  "StreamNameCondition": {
    "ComparisonOperator": "string",
    "ComparisonValue": "string"
  }
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

MaxResults

Il numero massimo di stream da restituire nella risposta. Il valore predefinito è 10.000.

Tipo: integer

Intervallo valido: valore minimo di 1. Valore massimo pari a 10000.

Campo obbligatorio: no

NextToken

Se si specifica questo parametro, quando il risultato di un'`ListStreams` operazione viene troncato, la chiamata restituisce il `NextToken` valore nella risposta. Per ricevere un altro batch di stream, fornisci questo token nella tua prossima richiesta.

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: [a-zA-Z0-9+/=]*

Campo obbligatorio: no

StreamNameCondition

Facoltativo: restituisce solo gli stream che soddisfano una condizione specifica. Attualmente, puoi specificare solo il prefisso del nome di uno stream come condizione.

Tipo: oggetto [StreamNameCondition](#)

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "NextToken": "string",
  "StreamInfoList": [
    {
      "CreationTime": number,
      "DataRetentionInHours": number,
      "DeviceName": "string",
      "KmsKeyId": "string",
      "MediaType": "string",
      "Status": "string",
      "StreamARN": "string",
      "StreamName": "string",
      "Version": "string"
    }
  ]
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

NextToken

Se la risposta viene troncata, la chiamata restituisce questo elemento con un token. Per ottenere il prossimo batch di stream, usa questo token nella tua prossima richiesta.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: `[a-zA-Z0-9+/=]*`

StreamInfoList

Un array di oggetti `StreamInfo`.

Tipo: matrice di oggetti [StreamInfo](#)

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)

- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

ListTagsForResource

Servizio: Amazon Kinesis Video Streams

Restituisce un elenco di tag associati al canale di segnalazione specificato.

Sintassi della richiesta

```
POST /ListTagsForResource HTTP/1.1
Content-type: application/json

{
  "NextToken": "string",
  "ResourceARN": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

[NextToken](#)

Se si specifica questo parametro e il risultato di una `ListTagsForResource` chiamata viene troncato, la risposta include un token che è possibile utilizzare nella richiesta successiva per recuperare il successivo batch di tag.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: `[a-zA-Z0-9+/=]*`

Campo obbligatorio: no

[ResourceARN](#)

L'Amazon Resource Name (ARN) del canale di segnalazione per il quale desideri elencare i tag.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: sì

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "NextToken": "string",
  "Tags": {
    "string" : "string"
  }
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

[NextToken](#)

Se specifichi questo parametro e il risultato di una `ListTagsForResource` chiamata viene troncato, la risposta include un token che puoi utilizzare nella richiesta successiva per recuperare il set di tag successivo.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: `[a-zA-Z0-9+/=]*`

[Tags](#)

Una mappa delle chiavi e dei valori dei tag associati al canale di segnalazione specificato.

Tipo: mappatura stringa a stringa

Voci sulla mappa: numero massimo di 50 elementi.

Limitazioni di lunghezza della chiave: la lunghezza minima è 1. La lunghezza massima è 128 caratteri.

Modello di chiave: $^{\wedge}([\backslash\{L\}\backslash\{Z\}\backslash\{N\}_\cdot : / = + \backslash - @] *) \$$

Limiti di lunghezza del valore: lunghezza minima di 0. La lunghezza massima è 256 caratteri.

Modello di valore: $[\backslash\{L\}\backslash\{Z\}\backslash\{N\}_\cdot : / = + \backslash - @] *$

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)

- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

ListTagsForStream

Servizio: Amazon Kinesis Video Streams

Restituisce un elenco di tag associati allo stream specificato.

Nella richiesta, è necessario specificare il `StreamName` o il `StreamARN`.

Sintassi della richiesta

```
POST /listTagsForStream HTTP/1.1
Content-type: application/json
```

```
{
  "NextToken": "string",
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

NextToken

Se si specifica questo parametro e il risultato di una `ListTagsForStream` chiamata viene troncato, la risposta include un token che è possibile utilizzare nella richiesta successiva per recuperare il successivo batch di tag.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: `[a-zA-Z0-9+/=]*`

Campo obbligatorio: no

StreamARN

L'Amazon Resource Name (ARN) dello stream per cui desideri elencare i tag.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome dello stream per cui desideri elencare i tag.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "NextToken": "string",
  "Tags": {
    "string" : "string"
  }
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

NextToken

Se specificate questo parametro e il risultato di una `ListTags` chiamata viene troncato, la risposta include un token che potete utilizzare nella richiesta successiva per recuperare il successivo set di tag.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 512 caratteri.

Modello: `[a-zA-Z0-9+/=]*`

Tags

Una mappa delle chiavi e dei valori dei tag associati allo stream specificato.

Tipo: mappatura stringa a stringa

Voci sulla mappa: numero massimo di 50 elementi.

Limitazioni di lunghezza della chiave: la lunghezza minima è 1. La lunghezza massima è 128 caratteri.

Modello di chiave: `^([\p{L}\p{Z}\p{N}_ . :/=+\-@]*)$`

Limiti di lunghezza del valore: lunghezza minima di 0. La lunghezza massima è 256 caratteri.

Modello di valore: `[\p{L}\p{Z}\p{N}_ . :/=+\-@]*`

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

InvalidResourceFormatException

Il formato di `StreamARN` non è valido.

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire questa operazione.

Codice di stato HTTP: 401

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

StartEdgeConfigurationUpdate

Servizio: Amazon Kinesis Video Streams

Un'API asincrona che aggiorna la configurazione edge esistente di uno stream. Kinesis Video Stream sincronizzerà la configurazione edge dello stream con il componente Edge Agent IoT Greengrass che funziona su un dispositivo Hub IoT, configurato presso la tua sede. Il tempo di sincronizzazione può variare e dipende dalla connettività del dispositivo Hub. SyncStatusVerrà aggiornato non appena la configurazione edge verrà riconosciuta e sincronizzata con Edge Agent.

Se questa API viene richiamata per la prima volta, verrà creata una nuova configurazione edge per lo stream e lo stato di sincronizzazione verrà impostato su. SYNCING Dovrai attendere che lo stato di sincronizzazione raggiunga uno stato terminale come:IN_SYNC, oppureSYNC_FAILED, prima di utilizzare nuovamente questa API. Se richiami questa API durante il processo di sincronizzazione, ResourceInUseException verrà generato un. La connettività della configurazione edge dello stream e dell'Edge Agent verrà ritentata per 15 minuti. Dopo 15 minuti, lo stato passerà allo SYNC_FAILED stato.

Per spostare una configurazione di bordo da un dispositivo a un altro, utilizzare [DeleteEdgeConfiguration](#) per eliminare la configurazione di bordo corrente. È quindi possibile richiamare StartEdgeConfigurationUpdate con un ARN del dispositivo Hub aggiornato.

Note

Questa API non è disponibile nella regione AWS Africa (Città del Capo), af-south-1.

Sintassi della richiesta

```
POST /startEdgeConfigurationUpdate HTTP/1.1
Content-type: application/json

{
  "EdgeConfig": {
    "DeletionConfig": {
      "DeleteAfterUpload": boolean,
      "EdgeRetentionInHours": number,
      "LocalSizeConfig": {
        "MaxLocalMediaSizeInMB": number,
        "StrategyOnFullSize": "string"
      }
    }
  }
}
```

```

    },
    "HubDeviceArn": "string",
    "RecorderConfig": {
      "MediaSourceConfig": {
        "MediaUriSecretArn": "string",
        "MediaUriType": "string"
      },
      "ScheduleConfig": {
        "DurationInSeconds": number,
        "ScheduleExpression": "string"
      }
    },
    "UploaderConfig": {
      "ScheduleConfig": {
        "DurationInSeconds": number,
        "ScheduleExpression": "string"
      }
    }
  },
  "StreamARN": "string",
  "StreamName": "string"
}

```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

[EdgeConfig](#)

I dettagli di configurazione dell'edge necessari per richiamare il processo di aggiornamento.

Tipo: oggetto [EdgeConfig](#)

Campo obbligatorio: sì

[StreamARN](#)

Il nome della risorsa Amazon (ARN) del flusso. Specificare il StreamName o il StreamARN

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome dello stream di cui si desidera aggiornare la configurazione edge. Specificate il StreamName o ilStreamARN.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "CreationTime": number,
  "EdgeConfig": {
    "DeletionConfig": {
      "DeleteAfterUpload": boolean,
      "EdgeRetentionInHours": number,
      "LocalSizeConfig": {
        "MaxLocalMediaSizeInMB": number,
        "StrategyOnFullSize": "string"
      }
    },
    "HubDeviceArn": "string",
    "RecorderConfig": {
      "MediaSourceConfig": {
        "MediaUriSecretArn": "string",
        "MediaUriType": "string"
      },
      "ScheduleConfig": {
        "DurationInSeconds": number,
```

```

        "ScheduleExpression": "string"
    }
},
"UploaderConfig": {
    "ScheduleConfig": {
        "DurationInSeconds": number,
        "ScheduleExpression": "string"
    }
}
},
"FailedStatusDetails": "string",
"LastUpdatedTime": number,
"StreamARN": "string",
"StreamName": "string",
"SyncStatus": "string"
}

```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

CreationTime

Il timestamp in cui è stata creata per la prima volta la configurazione edge di uno stream.

Tipo: Timestamp

EdgeConfig

Una descrizione della configurazione edge dello stream che verrà utilizzata per la sincronizzazione con il componente Edge Agent IoT Greengrass. Il componente Edge Agent verrà eseguito su un dispositivo Hub IoT configurato presso la tua sede.

Tipo: oggetto [EdgeConfig](#)

FailedStatusDetails

Una descrizione dello stato di errore generato.

-Tipo: stringa

LastUpdatedTime

Il timestamp in cui la configurazione edge di uno stream è stata aggiornata l'ultima volta.

Tipo: Timestamp

StreamARN

Il nome della risorsa Amazon (ARN) del flusso.

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

StreamName

Il nome dello stream da cui è stata aggiornata la configurazione edge.

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

SyncStatus

Lo stato di sincronizzazione corrente della configurazione edge dello stream. Quando richiami questa API, lo stato di sincronizzazione verrà impostato sullo SYNCING stato. Utilizza l'DescribeEdgeConfigurationAPI per ottenere lo stato più recente della configurazione edge.

•Tipo: stringa

Valori validi: SYNCING | ACKNOWLEDGED | IN_SYNC | SYNC_FAILED | DELETING | DELETE_FAILED | DELETING_ACKNOWLEDGED

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

NoDataRetentionException

La conservazione dei dati Stream in ore è pari a zero.

Codice di stato HTTP: 400

ResourceInUseException

Quando l'ingresso StreamARN o ChannelARN l'ingresso CLOUD_STORAGE_MODE sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'StreamARN ingresso fornito ChannelARN o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'DescribeMediaStorageConfigurationAPI per determinare a cosa è mappato lo stream di un determinato canale.
2. L'DescribeMappedResourceConfigurationAPI per determinare il canale su cui è mappato lo stream specificato.
3. L'DescribeStream o l'DescribeSignalingChannelAPI per determinare lo stato della risorsa.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

TagResource

Servizio: Amazon Kinesis Video Streams

Aggiunge uno o più tag a un canale di segnalazione. Un tag è una coppia chiave-valore (il valore è facoltativo) che è possibile definire e assegnare alle risorse. AWS Se si specifica un tag già esistente, il valore del tag viene sostituito con il valore specificato nella richiesta. Per ulteriori informazioni, vedere [Utilizzo dei tag di allocazione dei costi](#) nella Guida per l'utente di AWS Billing and Cost Management and Cost Management.

Sintassi della richiesta

```
POST /TagResource HTTP/1.1
Content-type: application/json

{
  "ResourceARN": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

[ResourceARN](#)

L'Amazon Resource Name (ARN) del canale di segnalazione a cui desideri aggiungere i tag.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+

Campo obbligatorio: sì

Tags

Un elenco di tag da associare al canale di segnalazione specificato. Ogni tag è una coppia chiave-valore.

Tipo: matrice di oggetti [Tag](#)

Membri dell'array: numero minimo di 1 elemento. Numero massimo di 50 item.

Campo obbligatorio: sì

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

TagsPerResourceExceededLimitException

Hai superato il limite di tag che puoi associare alla risorsa. Uno stream video Kinesis può supportare fino a 50 tag.

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

TagStream

Servizio: Amazon Kinesis Video Streams

Aggiunge uno o più tag a uno stream. Un tag è una coppia chiave-valore (il valore è facoltativo) che puoi definire e assegnare alle risorse. AWS Se si specifica un tag già esistente, il valore del tag viene sostituito con il valore specificato nella richiesta. Per ulteriori informazioni, vedere [Utilizzo dei tag di allocazione dei costi](#) nella Guida per l'utente di AWS Billing and Cost Management and Cost Management.

È necessario fornire il `StreamName` o il `StreamARN`.

Questa operazione richiede l'autorizzazione per l'operazione `KinesisVideo:TagStream`.

Uno stream video Kinesis può supportare fino a 50 tag.

Sintassi della richiesta

```
POST /tagStream HTTP/1.1
Content-type: application/json

{
  "StreamARN": "string",
  "StreamName": "string",
  "Tags": {
    "string" : "string"
  }
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

StreamARN

L'Amazon Resource Name (ARN) della risorsa a cui desideri aggiungere il tag o i tag.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome dello stream a cui desideri aggiungere il tag o i tag.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Tags

Un elenco di tag da associare allo stream specificato. Ogni tag è una coppia chiave-valore (il valore è opzionale).

Tipo: mappatura stringa a stringa

Voci sulla mappa: numero massimo di 50 elementi.

Limitazioni di lunghezza della chiave: la lunghezza minima è 1. La lunghezza massima è 128 caratteri.

Modello di chiave: `^([\p{L}\p{Z}\p{N}_.: / = + \ - @] *) $`

Limiti di lunghezza del valore: lunghezza minima di 0. La lunghezza massima è 256 caratteri.

Modello di valore: `[\p{L}\p{Z}\p{N}_.: / = + \ - @] *`

Campo obbligatorio: sì

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

InvalidResourceFormatException

Il formato di `StreamARN` non è valido.

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire questa operazione.

Codice di stato HTTP: 401

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

TagsPerResourceExceededLimitException

Hai superato il limite di tag che puoi associare alla risorsa. Uno stream video Kinesis può supportare fino a 50 tag.

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

UntagResource

Servizio: Amazon Kinesis Video Streams

Rimuove uno o più tag da un canale di segnalazione. Nella richiesta, specificate solo una o più chiavi di tag; non specificate il valore. Se specificate una chiave di tag che non esiste, viene ignorata.

Sintassi della richiesta

```
POST /UntagResource HTTP/1.1
Content-type: application/json

{
  "ResourceARN": "string",
  "TagKeyList": [ "string" ]
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

[ResourceARN](#)

L'Amazon Resource Name (ARN) del canale di segnalazione da cui desideri rimuovere i tag.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: sì

[TagKeyList](#)

Un elenco delle chiavi dei tag che desideri rimuovere.

Tipo: matrice di stringhe

Membri dell'array: numero minimo di 1 elemento. Numero massimo di 50 item.

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: $^([\backslash\{L\}\backslash\{Z\}\backslash\{N\}_\cdot :/=+\backslash-@]^*)\$$

Campo obbligatorio: sì

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

UntagStream

Servizio: Amazon Kinesis Video Streams

Rimuove uno o più tag da uno stream. Nella richiesta, specificate solo una o più chiavi di tag; non specificate il valore. Se specificate una chiave di tag che non esiste, viene ignorata.

Nella richiesta, è necessario fornire `StreamName` o `StreamARN`.

Sintassi della richiesta

```
POST /untagStream HTTP/1.1
Content-type: application/json

{
  "StreamARN": "string",
  "StreamName": "string",
  "TagKeyList": [ "string" ]
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

StreamARN

L'Amazon Resource Name (ARN) dello stream da cui desideri rimuovere i tag.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome dello stream da cui desideri rimuovere i tag.

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

TagKeyList

Un elenco delle chiavi dei tag che vuoi rimuovere.

Tipo: matrice di stringhe

Membri dell'array: numero minimo di 1 elemento. Numero massimo di 50 item.

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

Campo obbligatorio: sì

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

`InvalidResourceFormatException`

Il formato di `StreamARN` non è valido.

Codice di stato HTTP: 400

`NotAuthorizedException`

Il chiamante non è autorizzato a eseguire questa operazione.

Codice di stato HTTP: 401

`ResourceNotFoundException`

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

UpdateDataRetention

Servizio: Amazon Kinesis Video Streams

Aumenta o diminuisce il periodo di conservazione dei dati dello stream in base al valore specificato. Per indicare se desideri aumentare o diminuire il periodo di conservazione dei dati, specifica il `Operation` parametro nel corpo della richiesta. Nella richiesta, è necessario specificare il `StreamName` o il `StreamARN`.

Questa operazione richiede l'autorizzazione per l'operazione `KinesisVideo:UpdateDataRetention`.

La modifica del periodo di conservazione dei dati influisce sui dati nello stream nel modo seguente:

- Se il periodo di conservazione dei dati viene aumentato, i dati esistenti vengono conservati per il nuovo periodo di conservazione. Ad esempio, se il periodo di conservazione dei dati viene aumentato da un'ora a sette ore, tutti i dati esistenti vengono conservati per sette ore.
- Se il periodo di conservazione dei dati viene ridotto, i dati esistenti vengono conservati per il nuovo periodo di conservazione. Ad esempio, se il periodo di conservazione dei dati viene ridotto da sette ore a un'ora, tutti i dati esistenti vengono conservati per un'ora e tutti i dati più vecchi di un'ora vengono eliminati immediatamente.

Sintassi della richiesta

```
POST /updateDataRetention HTTP/1.1
Content-type: application/json

{
  "CurrentVersion": "string",
  "DataRetentionChangeInHours": number,
  "Operation": "string",
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

CurrentVersion

La versione dello stream di cui desideri modificare il periodo di conservazione. Per ottenere la versione, chiama l'API `DescribeStream` o l'`ListStreamsAPI`.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 64 caratteri.

Modello: `[a-zA-Z0-9]+`

Campo obbligatorio: sì

DataRetentionChangeInHours

Il numero di ore in base al quale regolare la conservazione corrente. Il valore specificato viene aggiunto o sottratto dal valore corrente, a seconda di `operation`

Il valore minimo per la conservazione dei dati è 0 e il valore massimo è 87600 (dieci anni).

Tipo: integer

Intervallo valido: valore minimo di 1.

Campo obbligatorio: sì

Operation

Indica se si desidera aumentare o diminuire il periodo di conservazione.

▪Tipo: stringa

Valori validi: `INCREASE_DATA_RETENTION` | `DECREASE_DATA_RETENTION`

Campo obbligatorio: sì

StreamARN

L'Amazon Resource Name (ARN) dello stream di cui desideri modificare il periodo di conservazione.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome dello stream di cui desideri modificare il periodo di conservazione.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire questa operazione.

Codice di stato HTTP: 401

ResourceInUseException

Quando l'ingresso StreamARN o ChannelARN l'ingresso CLOUD_STORAGE_MODE sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'StreamARNingresso fornito ChannelARN o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'DescribeMediaStorageConfigurationAPI per determinare a cosa è mappato lo stream di un determinato canale.
2. L'DescribeMappedResourceConfigurationAPI per determinare il canale su cui è mappato lo stream specificato.
3. L'DescribeStreamo l'DescribeSignalingChannelAPI per determinare lo stato della risorsa.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

VersionMismatchException

La versione dello stream che hai specificato non è la versione più recente. Per ottenere la versione più recente, usa l'[DescribeStreamAPI](#).

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)

- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

UpdateImageGenerationConfiguration

Servizio: Amazon Kinesis Video Streams

Aggiorna i ImageProcessingConfiguration campi StreamInfo and.

Sintassi della richiesta

```
POST /updateImageGenerationConfiguration HTTP/1.1
Content-type: application/json
```

```
{
  "ImageGenerationConfiguration": {
    "DestinationConfig": {
      "DestinationRegion": "string",
      "Uri": "string"
    },
    "Format": "string",
    "FormatConfig": {
      "string" : "string"
    },
    "HeightPixels": number,
    "ImageSelectorType": "string",
    "SamplingInterval": number,
    "Status": "string",
    "WidthPixels": number
  },
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ImageGenerationConfiguration

La struttura che contiene le informazioni necessarie per la consegna delle immagini KVS. Se la struttura è nulla, la configurazione verrà eliminata dallo stream.

Tipo: oggetto [ImageGenerationConfiguration](#)

Campo obbligatorio: no

[StreamARN](#)

L'Amazon Resource Name (ARN) del flusso video Kinesis da cui desideri aggiornare la configurazione di generazione dell'immagine. È necessario specificare il `StreamName` o il `StreamARN`

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

[StreamName](#)

Il nome dello stream da cui aggiornare la configurazione di generazione dell'immagine. È necessario specificare il `StreamName` o il `StreamARN`.

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non si dispone delle autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

NoDataRetentionException

La conservazione dei dati Stream in ore è pari a zero.

Codice di stato HTTP: 400

ResourceInUseException

Quando l'ingresso `StreamARN` o `ChannelARN` l'ingresso `CLOUD_STORAGE_MODE` sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'`StreamARN` ingresso fornito `ChannelARN` o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'`DescribeMediaStorageConfigurationAPI` per determinare a cosa è mappato lo stream di un determinato canale.
2. L'`DescribeMappedResourceConfigurationAPI` per determinare il canale su cui è mappato lo stream specificato.
3. L'`DescribeStream` o l'`DescribeSignalingChannelAPI` per determinare lo stato della risorsa.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

UpdateMediaStorageConfiguration

Servizio: Amazon Kinesis Video Streams

`SignalingChannel` associa a a uno stream per archiviare i file multimediali. È possibile specificare due modalità di segnalazione:

- Se `StorageStatus` è abilitata, i dati verranno archiviati nell'`StreamARN` apposito spazio. Affinché WebRTC Ingestion funzioni, lo stream deve avere la conservazione dei dati abilitata.
- Se `StorageStatus` è disabilitato, non verrà archiviato alcun dato e il `StreamARN` parametro non sarà necessario.

Important

Se `StorageStatus` è abilitato, le connessioni dirette peer-to-peer (master-viewer) non si verificano più. I peer si connettono direttamente alla sessione di archiviazione. È necessario chiamare `JoinStorageSessionAPI` per attivare l'invio di un'offerta SDP e stabilire una connessione tra un peer e la sessione di archiviazione.

Sintassi della richiesta

```
POST /updateMediaStorageConfiguration HTTP/1.1
Content-type: application/json
```

```
{
  "ChannelARN": "string",
  "MediaStorageConfiguration": {
    "Status": "string",
    "StreamARN": "string"
  }
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ChannelARN

L'Amazon Resource Name (ARN) del canale.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: sì

MediaStorageConfiguration

Una struttura che incapsula o contiene le proprietà di configurazione dello storage multimediale.

Tipo: oggetto [MediaStorageConfiguration](#)

Campo obbligatorio: sì

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non disponete delle autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

NoDataRetentionException

La conservazione dei dati Stream in ore è pari a zero.

Codice di stato HTTP: 400

ResourceInUseException

Quando l'ingresso StreamARN o ChannelARN l'ingresso CLOUD_STORAGE_MODE sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'StreamARNingresso fornito ChannelARN o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'DescribeMediaStorageConfigurationAPI per determinare a cosa è mappato lo stream di un determinato canale.
2. L'DescribeMappedResourceConfigurationAPI per determinare il canale su cui è mappato lo stream specificato.
3. L'DescribeStreamo l'DescribeSignalingChannelAPI per determinare lo stato della risorsa.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)
- [AWS SDK per C++](#)

- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

UpdateNotificationConfiguration

Servizio: Amazon Kinesis Video Streams

Aggiorna le informazioni di notifica per uno stream.

Sintassi della richiesta

```
POST /updateNotificationConfiguration HTTP/1.1
Content-type: application/json

{
  "NotificationConfiguration": {
    "DestinationConfig": {
      "Uri": "string"
    },
    "Status": "string"
  },
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

NotificationConfiguration

La struttura contenente le informazioni richieste per le notifiche. Se la struttura è nulla, la configurazione verrà eliminata dallo stream.

Tipo: oggetto [NotificationConfiguration](#)

Campo obbligatorio: no

StreamARN

L'Amazon Resource Name (ARN) del flusso video Kinesis da cui desideri aggiornare la configurazione delle notifiche. Devi specificare il StreamName o il StreamARN

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome dello stream da cui aggiornare la configurazione delle notifiche. È necessario specificare il `StreamName` o il `StreamARN`.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non si dispone delle autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

NoDataRetentionException

La conservazione dei dati Stream in ore è pari a zero.

Codice di stato HTTP: 400

ResourceInUseException

Quando l'ingresso StreamARN o ChannelARN l'ingresso CLOUD_STORAGE_MODE sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'StreamARNingresso fornito ChannelARN o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'DescribeMediaStorageConfigurationAPI per determinare a cosa è mappato lo stream di un determinato canale.
2. L'DescribeMappedResourceConfigurationAPI per determinare il canale su cui è mappato lo stream specificato.
3. L'DescribeStreamo l'DescribeSignalingChannelAPI per determinare lo stato della risorsa.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)

- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

UpdateSignalingChannel

Servizio: Amazon Kinesis Video Streams

Aggiorna il canale di segnalazione esistente. Si tratta di un'operazione asincrona che richiede tempo per essere completata.

Se il `MessageTtlSeconds` valore viene aggiornato (aumentato o ridotto), si applica solo ai nuovi messaggi inviati tramite questo canale dopo l'aggiornamento. I messaggi esistenti sono ancora scaduti in base al `MessageTtlSeconds` valore precedente.

Sintassi della richiesta

```
POST /updateSignalingChannel HTTP/1.1
Content-type: application/json

{
  "ChannelARN": "string",
  "CurrentVersion": "string",
  "SingleMasterConfiguration": {
    "MessageTtlSeconds": number
  }
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ChannelARN

L'Amazon Resource Name (ARN) del canale di segnalazione che desideri aggiornare.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: sì

CurrentVersion

La versione corrente del canale di segnalazione che desideri aggiornare.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 64 caratteri.

Modello: [a-zA-Z0-9]+

Campo obbligatorio: sì

SingleMasterConfiguration

La struttura contenente la configurazione per il SINGLE_MASTER tipo di canale di segnalazione che si desidera aggiornare.

Tipo: oggetto [SingleMasterConfiguration](#)

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non hai le autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 401

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceInUseException

Quando l'ingresso StreamARN o ChannelARN l'ingresso CLOUD_STORAGE_MODE sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'StreamARNingresso fornito ChannelARN o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'DescribeMediaStorageConfigurationAPI per determinare a cosa è mappato lo stream di un determinato canale.
2. L'DescribeMappedResourceConfigurationAPI per determinare il canale su cui è mappato lo stream specificato.
3. L'DescribeStreamo l'DescribeSignalingChannelAPI per determinare lo stato della risorsa.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

VersionMismatchException

La versione dello stream che hai specificato non è la versione più recente. Per ottenere la versione più recente, usa l'[DescribeStream](#)API.

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)

- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

UpdateStream

Servizio: Amazon Kinesis Video Streams

Aggiorna i metadati dello streaming, come il nome del dispositivo e il tipo di supporto.

Devi fornire il nome dello stream o l'Amazon Resource Name (ARN) dello stream.

Per assicurarti di disporre della versione più recente dello stream prima di aggiornarlo, puoi specificare la versione dello stream. Kinesis Video Streams assegna una versione a ogni stream. Quando aggiorni uno stream, Kinesis Video Streams assegna un nuovo numero di versione. Per ottenere la versione di streaming più recente, utilizza l'API. [DescribeStream](#)

UpdateStream è un'operazione asincrona e il suo completamento richiede tempo.

Sintassi della richiesta

```
POST /updateStream HTTP/1.1
Content-type: application/json

{
  "CurrentVersion": "string",
  "DeviceName": "string",
  "MediaType": "string",
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

CurrentVersion

La versione dello stream di cui desideri aggiornare i metadati.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 64 caratteri.

Modello: [a-zA-Z0-9]+

Campo obbligatorio: sì

DeviceName

Il nome del dispositivo che sta scrivendo nello stream.

Note

Nell'implementazione corrente, Kinesis Video Streams non utilizza questo nome.

─Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

MediaType

Il tipo di file multimediale dello stream. Utilizzato `MediaType` per specificare il tipo di contenuto che lo stream contiene per i consumatori dello stream. Per ulteriori informazioni sui tipi di file multimediali, consulta [Tipi di file multimediali](#). Se scegli di specificare il `MediaType`, vedi [Requisiti di denominazione](#).

Per riprodurre video sulla console, è necessario specificare il tipo di video corretto. Ad esempio, se il video nello stream è H.264, specifica `video/h264` come `MediaType`

─Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: [\w\-\.\.]+/[\w\-\.\.]+(,[\w\-\.\.]+/[\w\-\.\.]+)*

Campo obbligatorio: no

StreamARN

L'ARN dello stream di cui desideri aggiornare i metadati.

─Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome dello stream di cui desideri aggiornare i metadati.

Il nome dello stream è un identificatore dello stream e deve essere univoco per ogni account e regione.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire questa operazione.

Codice di stato HTTP: 401

ResourceInUseException

Quando l'ingresso `StreamARN` o `ChannelARN` l'ingresso `CLOUD_STORAGE_MODE` sono già mappati su un'altra risorsa Kinesis Video Stream, o se l'`StreamARN` ingresso fornito `ChannelARN` o non è nello stato Attivo, prova una delle seguenti soluzioni:

1. L'`DescribeMediaStorageConfigurationAPI` per determinare a cosa è mappato lo stream di un determinato canale.
2. L'`DescribeMappedResourceConfigurationAPI` per determinare il canale su cui è mappato lo stream specificato.
3. L'`DescribeStream` o l'`DescribeSignalingChannelAPI` per determinare lo stato della risorsa.

Codice di stato HTTP: 400

ResourceNotFoundException

Amazon Kinesis Video Streams non riesce a trovare lo stream specificato.

Codice di stato HTTP: 404

VersionMismatchException

La versione dello stream che hai specificato non è la versione più recente. Per ottenere la versione più recente, usa l'[DescribeStreamAPI](#).

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)

- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

Amazon Kinesis Video Streams

Le seguenti azioni sono supportate da Amazon Kinesis Video Streams Media:

- [GetMedia](#)
- [PutMedia](#)

GetMedia

Servizio: Amazon Kinesis Video Streams Media

Usa questa API per recuperare contenuti multimediali da un flusso video Kinesis. Nella richiesta, identifichi il nome dello stream o Amazon Resource Name (ARN) dello stream e il blocco iniziale. Kinesis Video Streams restituisce quindi un flusso di blocchi ordinati per numero di frammento.

Note

Devi prima chiamare l'`GetDataEndpointAPI` per ottenere un endpoint. Quindi invia le `GetMedia` richieste a questo endpoint utilizzando il parametro `--endpoint-url`.

Quando si inseriscono dati multimediali (frammenti) in uno stream, Kinesis Video Streams archivia ogni frammento in entrata e i relativi metadati in quello che viene chiamato «blocco». Per ulteriori informazioni [PutMedia](#), consulta. L'`GetMediaAPI` restituisce un flusso di questi blocchi a partire dal blocco specificato nella richiesta.

I seguenti limiti si applicano quando si utilizza l'API: `GetMedia`

- Un client può effettuare chiamate `GetMedia` fino a cinque volte al secondo per stream.
- Kinesis Video Streams invia dati multimediali a una velocità massima di 25 megabyte al secondo (o 200 megabit al secondo) durante una sessione. `GetMedia`

Note

Il codice di stato della risposta `GetMedia` HTTP verrà restituito immediatamente, ma la lettura del payload della risposta HTTP scadrà dopo 3 secondi se non ci sono frammenti importati disponibili per la riproduzione.

Note

Se viene generato un errore dopo aver richiamato un'API multimediale Kinesis Video Streams, oltre al codice di stato HTTP e al corpo della risposta, include le seguenti informazioni:

- `x-amz-ErrorType` intestazione HTTP: contiene un tipo di errore più specifico oltre a quello fornito dal codice di stato HTTP.
- `x-amz-RequestId` intestazione HTTP: se desideri segnalare un problema AWS, il team di supporto può diagnosticare meglio il problema se gli viene fornito il Request Id.

Sia il codice di stato HTTP che l' `ErrorType` intestazione possono essere utilizzati per prendere decisioni programmatiche sulla possibilità di correggere gli errori e in quali condizioni, oltre a fornire informazioni sulle azioni che il programmatore client potrebbe dover intraprendere per riprovare con successo.

[Per ulteriori informazioni, consulta la sezione Errori nella parte inferiore di questo argomento, oltre a Errori comuni.](#)

Sintassi della richiesta

```
POST /getMedia HTTP/1.1
Content-type: application/json

{
  "StartSelector": {
    "AfterFragmentNumber": "string",
    "ContinuationToken": "string",
    "StartSelectorType": "string",
    "StartTimestamp": number
  },
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

StartSelector

Identifica il blocco iniziale da ottenere dallo stream specificato.

Tipo: oggetto [StartSelector](#)

Campo obbligatorio: sì

[StreamARN](#)

L'ARN dello stream da cui desideri ricevere i contenuti multimediali. Se non si specifica `ilstreamARN`, è necessario specificare `ilstreamName`.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

[StreamName](#)

Il nome dello stream video Kinesis da cui desideri ottenere i contenuti multimediali. Se non si specifica `ilstreamName`, è necessario specificare `ilstreamARN`.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-Type: ContentType

Payload
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

La risposta restituisce le seguenti intestazioni HTTP.

ContentType

Il tipo di contenuto del supporto richiesto.

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: `^[a-zA-Z0-9_\.\-]+$`

La risposta restituisce quanto segue come corpo HTTP.

Payload

Il payload restituito da Kinesis Video Streams è una sequenza di blocchi del flusso specificato. Per ulteriori informazioni sui blocchi, consulta [PutMedia](#). I blocchi che Kinesis Video Streams restituisce nella chiamata includono anche i seguenti tag GetMedia Matroska (MKV) aggiuntivi:

- `AWS_KINESISVIDEO_CONTINUATION_TOKEN` (stringa UTF-8) - Nel caso in cui la GetMedia chiamata termini, puoi utilizzare questo token di continuazione nella tua richiesta successiva per ottenere il blocco successivo in cui è terminata l'ultima richiesta.
- `AWS_KINESISVIDEO_MILLIS_BEHIND_NOW` (stringa UTF-8) - Le applicazioni client possono utilizzare questo valore di tag per determinare quanto indietro il blocco restituito nella risposta si trova rispetto al blocco più recente dello stream.
- `AWS_KINESISVIDEO_FRAGMENT_NUMBER` - Numero di frammento restituito nel blocco.
- `AWS_KINESISVIDEO_SERVER_TIMESTAMP` - Timestamp del server del frammento.
- `AWS_KINESISVIDEO_PRODUCER_TIMESTAMP` - Timestamp del produttore del frammento.

I seguenti tag saranno presenti se si verifica un errore:

- `AWS_KINESISVIDEO_ERROR_CODE` - Descrizione in formato stringa di un errore che ha causato l'interruzione. GetMedia
- `AWS_KINESISVIDEO_ERROR_ID`: codice intero dell'errore.

I codici di errore sono i seguenti:

- 3002 - Errore durante la scrittura sullo stream
- 4000 - Il frammento richiesto non è stato trovato
- 4500 - Accesso negato per la chiave KMS dello stream
- 4501 - La chiave KMS di Stream è disabilitata
- 4502 - Errore di convalida sulla chiave KMS dello stream

- 4503 - La chiave KMS specificata nello stream non è disponibile
- 4504 - Utilizzo non valido della chiave KMS specificata nello stream
- 4505 - Stato non valido della chiave KMS specificata nello stream
- 4506 - Impossibile trovare la chiave KMS specificata nello stream
- 5000 - Errore interno

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

ConnectionLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di connessioni client consentite.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

InvalidEndpointException

Il chiamante ha utilizzato un endpoint errato per scrivere dati in uno stream. Quando riceve tale eccezione, l'utente deve chiamare `GetDataEndpoint` con `APIName` set to `PUT_MEDIA` e utilizzare l'endpoint from response per richiamare la chiamata successiva. `PutMedia`

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire un'operazione sullo stream specificato o il token è scaduto.

Codice di stato HTTP: 401

ResourceNotFoundException

Codice di stato: 404, Lo stream con il nome specificato non esiste.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

PutMedia

Servizio: Amazon Kinesis Video Streams Media

Usa questa API per inviare dati multimediali a un flusso video Kinesis.

Note

Devi prima chiamare l'GetDataEndpointAPI per ottenere un endpoint. Quindi invia le PutMedia richieste a questo endpoint utilizzando il parametro [--endpoint-url](#).

Nella richiesta, si utilizzano le intestazioni HTTP per fornire informazioni sui parametri, ad esempio il nome dello stream, il timestamp e se il valore del timestamp è assoluto o relativo a quando il produttore ha iniziato la registrazione. Utilizzate il corpo della richiesta per inviare i dati multimediali. Kinesis Video Streams supporta solo il formato contenitore Matroska (MKV) per l'invio di dati multimediali utilizzando questa API.

Sono disponibili le seguenti opzioni per l'invio di dati utilizzando questa API:

- Invia dati multimediali in tempo reale: ad esempio, una telecamera di sicurezza può inviare fotogrammi in tempo reale man mano che li genera. Questo approccio riduce al minimo la latenza tra la registrazione video e i dati inviati via cavo. Questo viene definito produttore continuo. In questo caso, un'applicazione consumer può leggere lo stream in tempo reale o quando necessario.
- Invia dati multimediali offline (in batch): ad esempio, una body camera potrebbe registrare video per ore e archivarli sul dispositivo. Successivamente, quando colleghi la videocamera alla porta docking, la videocamera può avviare una PutMedia sessione per inviare dati a un flusso video Kinesis. In questo scenario, la latenza non è un problema.

Quando utilizzi questa API, tieni presente le seguenti considerazioni:

- È necessario specificare `streamName` o `streamARN`, ma non entrambi.
- Per poter riprodurre i file multimediali sulla console o tramite HLS, la traccia 1 di ogni frammento deve contenere video con codifica h.264, il CodeCid nei metadati del frammento deve essere «V_MPEG/ISO/AVC» e i metadati del frammento devono includere dati privati del codec h.264 in formato AVCC. Facoltativamente, la traccia 2 di ogni frammento deve contenere audio codificato AAC, il CodeCid nei metadati del frammento deve essere «A_AAC» e i metadati del frammento devono includere dati privati del codec AAC.

- Potrebbe essere più semplice utilizzare una singola sessione di lunga durata PutMedia e inviare un gran numero di frammenti di dati multimediali nel payload. Per ogni frammento ricevuto, Kinesis Video Streams invia uno o più riconoscimenti. Potenziali considerazioni sulla rete potrebbero impedirti di ricevere tutti questi riconoscimenti man mano che vengono generati.
- Potresti scegliere più PutMedia sessioni consecutive, ognuna con meno frammenti, per assicurarti di ricevere tutti i riconoscimenti dal servizio in tempo reale.

Note

Se invii dati allo stesso stream in più PutMedia sessioni simultanee, i frammenti multimediali vengono interlacciati nello stream. È necessario assicurarsi che ciò sia corretto nel proprio scenario applicativo.

Quando si utilizza l'PutMediaAPI, si applicano i seguenti limiti:

- Un client può effettuare chiamate PutMedia fino a cinque volte al secondo per stream.
- Un client può inviare fino a cinque frammenti al secondo per stream.
- Kinesis Video Streams legge i dati multimediali a una velocità massima di 12,5 MB/secondo o 100 Mbps durante una sessione. PutMedia

Nota i seguenti vincoli. In questi casi, Kinesis Video Streams invia la conferma dell'errore nella risposta.

- I frammenti con codici temporali superiori al limite massimo consentito e che contengono più di 50 MB di dati non sono consentiti.
- I frammenti contenenti più di tre tracce non sono consentiti. Ogni fotogramma di ogni frammento deve avere lo stesso numero di traccia di una delle tracce definite nell'intestazione del frammento. Inoltre, ogni frammento deve contenere almeno un fotogramma per ogni traccia definita nell'intestazione del frammento.
- Ogni frammento deve contenere almeno un fotogramma per ogni traccia definita nei metadati del frammento.
- Il timestamp del primo fotogramma in un frammento deve essere successivo all'ultimo timestamp del frammento precedente.

- Anche uno stream MKV contenente più di un segmento MKV o contenente elementi MKV non consentiti (come) genera la conferma dell'errore. `track*`

Kinesis Video Streams archivia ogni frammento in entrata e i relativi metadati in un cosiddetto «blocco». I metadati del frammento includono quanto segue:

- Le intestazioni MKV fornite all'inizio della richiesta `PutMedia`
- I seguenti metadati specifici di Kinesis Video Streams per il frammento:
 - `server_timestamp`- Data e ora in cui Kinesis Video Streams ha iniziato a ricevere il frammento.
 - `producer_timestamp`- Timestamp, quando il produttore ha iniziato a registrare il frammento. Kinesis Video Streams utilizza tre informazioni ricevute nella richiesta per calcolare questo valore.
 - Il valore del codice di tempo del frammento ricevuto nel corpo della richiesta insieme al frammento.
 - Due intestazioni di richiesta: `producerStartTimeStamp` (quando il produttore ha iniziato la registrazione) e `fragmentTimeCodeType` (se il codice di tempo del frammento nel payload è assoluto o relativo).

Kinesis Video Streams `producer_timestamp` calcola quindi il frammento nel modo seguente:

Se è relativo `fragmentTimeCodeType`, allora

`producer_timestamp` = `producerStartTimeStamp` + codice temporale del frammento

Se `fragmentTimeCodeType` è assoluto, allora

`producer_timestamp` = codice di tempo del frammento (convertito in millisecondi)

- Numero di frammento univoco assegnato da Kinesis Video Streams.

Note

Quando effettui la `GetMedia` richiesta, Kinesis Video Streams restituisce un flusso di questi blocchi. Il client può elaborare i metadati secondo necessità.

Note

Questa operazione è disponibile solo per l' AWS SDK for Java. Non è supportata negli AWS SDK per altre lingue.

Note

Kinesis Video Streams non analizza e convalida i dati privati del codec durante l'ingestione e l'archiviazione tramite l'API. PutMedia KVS estrae e convalida le informazioni necessarie dai dati privati del codec per il confezionamento di frammenti MPEG-TS e MP4 quando utilizza lo streaming tramite le API HLS.

Note

Se viene generato un errore dopo aver richiamato un'API multimediale Kinesis Video Streams, oltre al codice di stato HTTP e al corpo della risposta, include le seguenti informazioni:

- `x-amz-ErrorType` intestazione HTTP: contiene un tipo di errore più specifico oltre a quello fornito dal codice di stato HTTP.
- `x-amz-RequestId` intestazione HTTP: se desideri segnalare un problema AWS, il team di supporto può diagnosticare meglio il problema se gli viene fornito il Request Id.

Sia il codice di stato HTTP che l' `ErrorType` intestazione possono essere utilizzati per prendere decisioni programmatiche sulla possibilità di correggere gli errori e in quali condizioni, oltre a fornire informazioni sulle azioni che il programmatore client potrebbe dover intraprendere per riprovare con successo.

[Per ulteriori informazioni, consulta la sezione Errori nella parte inferiore di questo argomento, oltre a Errori comuni.](#)

Sintassi della richiesta

```
POST /putMedia HTTP/1.1
x-amzn-stream-name: StreamName
```

```
x-amzn-stream-arn: StreamARN  
x-amzn-fragment-timecode-type: FragmentTimecodeType  
x-amzn-producer-start-timestamp: ProducerStartTimestamp
```

Payload

Parametri della richiesta URI

La richiesta utilizza i seguenti parametri URI.

[FragmentTimecodeType](#)

Questo valore viene passato come intestazione `x-amzn-fragment-timecode-type` HTTP.

Indica se i codici temporali nei frammenti (payload, corpo della richiesta HTTP) sono assoluti o relativi a `producerStartTimestamp`. Kinesis Video Streams utilizza queste informazioni per `producer_timestamp` calcolare il frammento ricevuto nella richiesta, come descritto nella panoramica dell'API.

Valori validi: ABSOLUTE | RELATIVE

Campo obbligatorio: sì

[ProducerStartTimestamp](#)

Si passa questo valore come intestazione HTTP. `x-amzn-producer-start-timestamp`

Questo è il timestamp del produttore in cui il produttore ha iniziato a registrare il file multimediale (non il timestamp dei frammenti specifici nella richiesta).

[StreamARN](#)

Si passa questo valore come intestazione HTTP. `x-amzn-stream-arn`

Amazon Resource Name (ARN) dello stream video Kinesis in cui desideri scrivere i contenuti multimediali. Se non specifichi il `streamARN`, devi specificare il `streamName`

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

[StreamName](#)

Questo valore viene passato come intestazione `x-amzn-stream-name` HTTP.

Nome dello stream video Kinesis in cui desideri scrivere i contenuti multimediali. Se non si specifica il `streamName`, è necessario specificare il `streamARN`.

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Corpo della richiesta

La richiesta accetta i seguenti dati binari.

[Payload](#)

I contenuti multimediali da scrivere nello stream video di Kinesis. Nell'implementazione attuale, Kinesis Video Streams supporta solo il formato contenitore Matroska (MKV) con un singolo segmento MKV. Un segmento può contenere uno o più cluster.

Note

Ogni cluster MKV viene mappato su un frammento di flusso video Kinesis. Qualunque sia la durata del cluster scelta, diventa la durata del frammento.

Sintassi della risposta

```
HTTP/1.1 200
```

Payload

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

La risposta restituisce quanto segue come corpo HTTP.

[Payload](#)

Dopo che Kinesis Video Streams PutMedia riceve correttamente una richiesta, il servizio convalida le intestazioni della richiesta. Il servizio inizia quindi a leggere il payload e invia prima una risposta HTTP 200.

Il servizio restituisce quindi uno stream contenente una serie di oggetti (Acknowledgementoggetti) JSON separati da nuove righe. I riconoscimenti vengono ricevuti sulla stessa connessione su cui vengono inviati i dati multimediali. Possono esserci molti riconoscimenti per una richiesta. PutMedia Ciascuno Acknowledgement è composto dalle seguenti coppie chiave-valore:

- **AckEventType**- Tipo di evento rappresentato dal riconoscimento.
 - **Buffering**: Kinesis Video Streams ha iniziato a ricevere il frammento. Kinesis Video Streams invia la prima conferma di buffering quando viene ricevuto il primo byte di dati frammentari.
 - **Ricevuto**: Kinesis Video Streams ha ricevuto l'intero frammento. Se non hai configurato lo stream per rendere persistenti i dati, il produttore può interrompere il buffering del frammento dopo aver ricevuto questa conferma.
 - **Persistente**: Kinesis Video Streams ha reso persistente il frammento (ad esempio, su Amazon S3). Ottieni questo riconoscimento se hai configurato lo stream per rendere persistenti i dati. Dopo aver ricevuto questo riconoscimento, il produttore può interrompere il buffering del frammento.
 - **Errore**: Kinesis Video Streams ha riscontrato un errore durante l'elaborazione del frammento. Puoi rivedere il codice di errore e determinare la prossima linea d'azione.
 - **Inattiva**: la PutMedia sessione è in corso. Tuttavia, Kinesis Video Streams al momento non riceve dati. Kinesis Video Streams invia questa conferma periodicamente per un massimo di 30 secondi dopo l'ultima ricezione dei dati. Se non viene ricevuto alcun dato entro 30 secondi, Kinesis Video Streams chiude la richiesta.

 Note

Questo riconoscimento può aiutare un produttore a determinare se la PutMedia connessione è attiva, anche se non invia alcun dato.

- **FragmentTimecode**- Codice temporale del frammento per il quale viene inviata la conferma.

L'elemento può mancare se è inattivo. **AckEventType**

- **FragmentNumber**- Numero del frammento generato da Kinesis Video Streams per il quale viene inviata la conferma.
- **ErrorId** **ErrorCode** - Se **AckEventType** è **Error**, questo campo fornisce il codice di errore corrispondente. Di seguito è riportato l'elenco degli ID di errore e dei relativi codici di errore e messaggi di errore:

- 4000 - STREAM_READ_ERROR - Errore durante la lettura del flusso di dati.
- 4001 - MAX_FRAGMENT_SIZE_REACHED - La dimensione del frammento è superiore al limite massimo consentito, 50 MB.
- 4002 - MAX_FRAGMENT_DURATION_REACHED - La durata del frammento è superiore al limite massimo consentito.
- 4003 - MAX_CONNECTION_DURATION_REACHED - La durata della connessione è superiore alla soglia massima consentita.
- 4004 - FRAGMENT_TIMECODE_LESSER_THAN_PREVIOUS - Il codice di tempo del frammento è inferiore al codice temporale precedente (all'interno di una chiamata, non è possibile inviare frammenti fuori ordine). PutMedia
- 4005 - MORE_THAN_ALLOWED_TRACKS_FOUND - In MKV è stata trovata più di una traccia. (deprecato)
- 4006 - INVALID_MKV_DATA - Impossibile analizzare il flusso di input come formato MKV valido.
- 4007 - INVALID_PRODUCER_TIMESTAMP - Timestamp del produttore non valido.
- 4008 - STREAM_NOT_ACTIVE - Lo stream non esiste più (eliminato).
- 4009 - FRAGMENT_METADATA_LIMIT_REACHED - Limite di metadati del frammento raggiunto. [Consulta la sezione Limiti della](#) guida per sviluppatori.
- 4010 - TRACK_NUMBER_MISMATCH - Il numero della traccia in un frame MKV non corrisponde alle tracce nell'intestazione MKV.
- 4011 - FRAMES_MISSING_FOR_TRACK - Il frammento non conteneva alcun frame per almeno una delle tracce nell'intestazione MKV.
- 4012 - INVALID_FRAGMENT_METADATA - Il nome dei metadati del frammento non può iniziare con la stringa `AWS_`.
- 4500 - KMS_KEY_ACCESS_DENIED - L'accesso alla chiave KMS specificata dallo stream è negato.
- 4501 - KMS_KEY_DISABLED - La chiave KMS specificata dallo stream è disabilitata.
- 4502 - KMS_KEY_VALIDATION_ERROR - La chiave KMS specificata nello stream non è riuscita.
- 4503 - KMS_KEY_UNAVAILABLE - La chiave KMS specificata dallo stream non è disponibile.
- 4504 - KMS_KEY_INVALID_USAGE - Utilizzo non valido della chiave KMS specificata dallo stream.

- 4505 - KMS_KEY_INVALID_STATE - La chiave KMS specificata dallo stream è in uno stato non valido.
- 4506 - KMS_KEY_NOT_FOUND - La chiave KMS specificata dallo stream non è stata trovata.
- 5000 - INTERNAL_ERROR - Errore interno del servizio.
- 5001 - ARCHIVAL_ERROR - Kinesis Video Streams non è riuscito a rendere persistenti i frammenti nell'archivio dati.

Note

Il produttore, mentre invia il payload per una richiesta di lunga PutMedia durata, dovrebbe leggere la risposta per i riconoscimenti. Un produttore potrebbe ricevere blocchi di riconoscimenti contemporaneamente, a causa del buffering su un server proxy intermedio. Un produttore che desidera ricevere riconoscimenti tempestivi può inviare meno frammenti per ogni richiesta. PutMedia

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

ConnectionLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di connessioni client consentite.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

InvalidEndpointException

Il chiamante ha utilizzato un endpoint errato per scrivere dati in uno stream. Quando riceve tale eccezione, l'utente deve chiamare `GetDataEndpoint` con `APIName` set to `PUT_MEDIA` e utilizzare l'endpoint from response per richiamare la chiamata successiva. `PutMedia`

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire un'operazione sullo stream specificato o il token è scaduto.

Codice di stato HTTP: 401

ResourceNotFoundException

Codice di stato: 404, Lo stream con il nome specificato non esiste.

Codice di stato HTTP: 404

Esempi

Formato di riconoscimento

Il formato del riconoscimento è il seguente:

```
{
  Acknowledgement : {
    "EventType": enum,
    "FragmentTimecode": Long,
    "FragmentNumber": Long,
    "ErrorId" : String
  }
}
```

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)

- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

Amazon Kinesis Video Streams

Le seguenti azioni sono supportate da Amazon Kinesis Video Streams Archived Media:

- [GetClip](#)
- [GetDASHStreamingSessionURL](#)
- [GetHLSStreamingSessionURL](#)
- [GetImages](#)
- [GetMediaForFragmentList](#)
- [ListFragments](#)

GetClip

Servizio: Amazon Kinesis Video Streams Archived Media

Scarica un file MP4 (clip) contenente i file multimediali su richiesta archiviati dal flusso video specificato nell'intervallo di tempo specificato.

Entrambi i parametri StreamName e StreamArn sono facoltativi, ma è necessario specificare il StreamName o StreamArn quando si richiama questa operazione API.

Note

È innanzitutto necessario chiamare l'GetDataEndpointAPI per ottenere un endpoint. Quindi invia le GetClip richieste a questo endpoint utilizzando il parametro [--endpoint-url](#).

Uno stream video di Amazon Kinesis presenta i seguenti requisiti per la fornitura di dati tramite MP4:

- Requisiti per la [riproduzione dei brani video](#).
- La conservazione dei dati deve essere superiore a 0.
- La traccia video di ogni frammento deve contenere i dati codec privati in Advanced Video Coding (AVC) per il formato H.264 e HEVC per il formato H.265. Per ulteriori informazioni, consulta [Specifica MPEG-4 ISO/IEC 14496-15](#). Per informazioni sull'adattamento dei dati di flusso a un dato formato, vedi i [contrassegni di adattamento NAL](#).
- La traccia audio (se presente) di ogni frammento deve contenere dati codec privati in formato AAC ([Specifica AAC ISO/IEC 13818-7](#)) o in formato [MS Wave](#).

Puoi monitorare la quantità di dati in uscita monitorando la CloudWatch metrica di GetClip.OutgoingBytes Amazon. Per informazioni sull'utilizzo per CloudWatch monitorare Kinesis Video Streams, [consulta Monitoring Kinesis Video Streams](#). [Per informazioni sui prezzi, consulta Prezzi e prezzi di Amazon Kinesis Video Streams AWS](#) . Si applicano i costi per i dati in uscita AWS .

Important

I dati privati del codec (CPD) contenuti in ogni frammento contengono informazioni di inizializzazione specifiche del codec, come la frequenza dei fotogrammi, la risoluzione e il profilo di codifica, necessarie per decodificare correttamente il frammento. Le modifiche CPD

non sono supportate tra i frammenti di destinazione della clip risultante. Il CPD deve rimanere coerente su tutti i supporti interrogati, altrimenti verrà restituito un errore.

⚠ Important

Le modifiche alla traccia non sono supportate. Le tracce devono rimanere coerenti su tutti i file multimediali interrogati. Viene restituito un errore se i frammenti dello stream passano da un formato contenente solo video ad avere sia audio che video, o se una traccia audio AAC viene sostituita da una traccia audio A-Law.

Sintassi della richiesta

```
POST /getClip HTTP/1.1
Content-type: application/json

{
  "ClipFragmentSelector": {
    "FragmentSelectorType": "string",
    "TimestampRange": {
      "EndTimeStamp": number,
      "StartTimeStamp": number
    }
  },
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ClipFragmentSelector

L'intervallo di tempo della clip richiesta e la fonte dei timestamp.

Tipo: oggetto ClipFragmentSelector

Campo obbligatorio: sì

StreamARN

L'Amazon Resource Name (ARN) dello stream per il quale recuperare il clip multimediale.

Devi specificare il StreamName o lo StreamArn.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome dello stream per il quale recuperare il clip multimediale.

È necessario specificare o StreamName StreamArn.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-Type: ContentType
```

```
Payload
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

La risposta restituisce le seguenti intestazioni HTTP.

ContentType

Il tipo di contenuto multimediale nella clip richiesta.

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: `^[a-zA-Z0-9_\.\\-]+$`

La risposta restituisce quanto segue come corpo HTTP.

Payload

File MP4 tradizionale che contiene il clip multimediale del flusso video specificato. L'output conterrà i primi 100 MB o i primi 200 frammenti del timestamp di inizio specificato. Per ulteriori informazioni, consulta [Kinesis Video Streams Limits](#).

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché hai superato un limite. Prova a effettuare la chiamata più tardi. Per informazioni sui limiti, consulta [Kinesis Video Streams Limits](#).

Codice di stato HTTP: 400

InvalidArgumentException

Un parametro specificato supera le restrizioni, non è supportato o non può essere utilizzato.

Codice di stato HTTP: 400

InvalidCodecPrivateDataException

I dati privati del codec in almeno una delle tracce del flusso video non sono validi per questa operazione.

Codice di stato HTTP: 400

InvalidMediaFrameException

Non è stato possibile analizzare uno o più fotogrammi nella clip richiesta in base al codec specificato.

Codice di stato HTTP: 400

MissingCodecPrivateDataException

Non sono stati trovati dati privati del codec in almeno una delle tracce del flusso video.

Codice di stato HTTP: 400

NoDataRetentionException

GetImages è stato richiesto per uno stream che non conserva dati (ovvero ha un valore `DataRetentionInHours` pari a 0).

Codice di stato HTTP: 400

NotAuthorizedException

Codice di stato: 403, il chiamante non è autorizzato a eseguire un'operazione sullo stream specificato o il token è scaduto.

Codice di stato HTTP: 401

ResourceNotFoundException

GetImages genererà questo errore quando Kinesis Video Streams non riesce a trovare lo stream specificato.

GetHLSStreamingSessionURL e GetDASHStreamingSessionURL genera questo errore se LIVE_REPLAY viene richiesta una sessione con un `PlaybackMode` di `ON_DEMAND` o per uno stream che non contiene frammenti nell'intervallo di tempo richiesto, o se LIVE viene richiesta una sessione con un `PlaybackMode` di per uno stream che non contiene frammenti negli ultimi 30 secondi.

Codice di stato HTTP: 404

UnsupportedStreamMediaTypeException

Il tipo di supporto (ad esempio, video h.264 o h.265 o audio AAC o G.711) non può essere determinato dagli ID dei codec delle tracce nel primo frammento di una sessione di riproduzione. L'ID del codec per la traccia 1 dovrebbe essere `V_MPEG/ISO/AVC` e, facoltativamente, l'ID del codec per la traccia 2 dovrebbe essere `A_AAC`.

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli SDK specifici della lingua, consulta quanto segue AWS :

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

GetDASHStreamingSessionURL

Servizio: Amazon Kinesis Video Streams Archived Media

Recupera un URL MPEG Dynamic Adaptive Streaming over HTTP (DASH) per lo stream. È quindi possibile aprire l'URL in un lettore multimediale per visualizzare i contenuti dello streaming.

StreamName e i parametri che i StreamARN parametri sono facoltativi, ma è necessario specificare StreamName o il StreamARN quando si richiama questa operazione API.

Uno stream video di Amazon Kinesis presenta i seguenti requisiti per la fornitura di dati tramite MPEG-DASH:

- Requisiti per la [riproduzione dei brani video](#).
- La conservazione dei dati deve essere superiore a 0.
- La traccia video di ogni frammento deve contenere i dati codec privati in Advanced Video Coding (AVC) per il formato H.264 e HEVC per il formato H.265. Per ulteriori informazioni, consulta [Specifica MPEG-4 ISO/IEC 14496-15](#). Per informazioni sull'adattamento dei dati di flusso a un dato formato, vedi i [contrassegni di adattamento NAL](#).
- La traccia audio (se presente) di ogni frammento deve contenere dati codec privati in formato AAC ([Specifica AAC ISO/IEC 13818-7](#)) o in formato [MS Wave](#).

La procedura seguente mostra come utilizzare MPEG-DASH con Kinesis Video Streams:

1. Chiama l'GetDataEndpointAPI per ottenere un endpoint. Quindi invia le GetDASHStreamingSessionURL richieste a questo endpoint utilizzando il parametro [--endpoint-url](#).
2. Recupera l'URL MPEG-DASH utilizzando. GetDASHStreamingSessionURL Kinesis Video Streams crea una sessione di streaming MPEG-DASH da utilizzare per accedere ai contenuti di uno stream utilizzando il protocollo MPEG-DASH. GetDASHStreamingSessionURL restituisce un URL autenticato (che include un token di sessione crittografato) per il manifesto MPEG-DASH della sessione (la risorsa principale necessaria per lo streaming con MPEG-DASH).

Note

Non condividere o archiviare questo token dove un'entità non autorizzata può accedervi. Il token fornisce l'accesso al contenuto dello stream. Proteggi il token con le stesse misure che usi con le tue AWS credenziali.

I file multimediali resi disponibili tramite il manifesto sono costituiti solo dallo stream, dall'intervallo di tempo e dal formato richiesti. Nessun altro dato multimediale (come frame esterni alla finestra richiesta o bitrate alternativi) viene reso disponibile.

3. Fornite l'URL (contenente il token di sessione crittografato) per il manifesto MPEG-DASH a un lettore multimediale che supporti il protocollo MPEG-DASH. Kinesis Video Streams rende disponibili il frammento di inizializzazione e i frammenti multimediali tramite l'URL del manifesto. Il frammento di inizializzazione contiene i dati privati del codec per lo stream e altri dati necessari per configurare il decodificatore e il renderer video o audio. I frammenti multimediali contengono fotogrammi video codificati o campioni audio codificati.
4. Il lettore multimediale riceve l'URL autenticato e richiede lo streaming di metadati e dati multimediali normalmente. Quando il lettore multimediale richiede dati, richiama le seguenti azioni:
 - **GetDashManifest**: recupera un manifesto MPEG DASH, che contiene i metadati per i file multimediali che si desidera riprodurre.
 - **GetMP4: InitFragment** recupera il frammento di inizializzazione MP4. Il lettore multimediale in genere carica il frammento di inizializzazione prima di caricare qualsiasi frammento multimediale. Questo frammento contiene gli atomi "fryp" e "moov" MP4 e gli atomi secondari necessari per inizializzare il decoder del lettore multimediale.

Il frammento di inizializzazione non corrisponde a un frammento in uno stream video Kinesis. Contiene solo i dati privati del codec per lo stream e la rispettiva traccia, necessari al lettore multimediale per decodificare i frame multimediali.

- **GetMP4MediaFragment**: recupera i frammenti multimediali MP4. Questi frammenti contengono gli atomi "" e "moofmdat" MP4 e i relativi atomi secondari, contenenti i frame multimediali del frammento codificato e i relativi timestamp.

Important

I dati privati del codec (CPD) contenuti in ogni frammento contengono informazioni di inizializzazione specifiche del codec, come la frequenza dei fotogrammi, la risoluzione e il profilo di codifica, necessarie per decodificare correttamente il frammento. Le modifiche CPD non sono supportate durante una sessione di streaming. Il CPD deve rimanere coerente attraverso i media interrogati.

⚠ Important

Le modifiche alla traccia non sono supportate. Le tracce devono rimanere coerenti su tutti i file multimediali interrogati. Lo streaming fallirà se i frammenti dello stream passano da un formato contenente solo video ad avere sia audio che video, o se una traccia audio AAC viene sostituita da una traccia audio A-Law.

I dati recuperati con questa azione sono fatturabili. Per maggiori dettagli, consulta la sezione [Prezzi](#).

ℹ Note

Per le restrizioni che si applicano alle sessioni MPEG-DASH, consulta Kinesis Video [Streams Limits](#).

Puoi monitorare la quantità di dati consumata dal lettore multimediale monitorando la CloudWatch metrica di `GetMP4MediaFragment.OutgoingBytes` Amazon. Per informazioni sull'utilizzo per CloudWatch monitorare Kinesis Video Streams, [consulta Monitoring Kinesis Video Streams](#). [Per informazioni sui prezzi, consulta Prezzi e prezzi di Amazon Kinesis Video Streams AWS](#) . Si applicano i costi sia per le sessioni HLS che per i dati in uscita AWS .

Per ulteriori informazioni su HLS, consulta [HTTP Live Streaming sul sito Apple Developer](#).

⚠ Important

Se viene generato un errore dopo aver richiamato un'API multimediale archiviata di Kinesis Video Streams, oltre al codice di stato HTTP e al corpo della risposta, include le seguenti informazioni:

- `x-amz-ErrorType` intestazione HTTP: contiene un tipo di errore più specifico oltre a quello fornito dal codice di stato HTTP.
- `x-amz-RequestId` intestazione HTTP: se desideri segnalare un problema al team di supporto, puoi diagnosticare meglio AWS il problema se fornisci il Request Id.

Sia il codice di stato HTTP che l' `ErrorType` intestazione possono essere utilizzati per prendere decisioni programmatiche sulla possibilità di riprovare gli errori e in quali condizioni, oltre a fornire informazioni sulle azioni che il programmatore del client potrebbe dover intraprendere per riprovare con successo.

[Per ulteriori informazioni, consulta la sezione Errori nella parte inferiore di questo argomento, nonché Errori comuni.](#)

Sintassi della richiesta

```
POST /getDASHStreamingSessionURL HTTP/1.1
Content-type: application/json
```

```
{
  "DASHFragmentSelector": {
    "FragmentSelectorType": "string",
    "TimestampRange": {
      "EndTimeStamp": number,
      "StartTimeStamp": number
    }
  },
  "DisplayFragmentNumber": "string",
  "DisplayFragmentTimestamp": "string",
  "Expires": number,
  "MaxManifestFragmentResults": number,
  "PlaybackMode": "string",
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

[DASHFragmentSelector](#)

L'intervallo di tempo del frammento richiesto e l'origine dei timestamp.

Questo parametro è obbligatorio se PlaybackMode è o. ON_DEMAND LIVE_REPLAY
Questo parametro è facoltativo se lo PlaybackMode è LIVE. Se PlaybackMode siLIVE, FragmentSelectorType può essere impostato, ma non TimestampRange deve essere impostato. Se PlaybackMode è ON_DEMAND oLIVE_REPLAY, entrambi FragmentSelectorType TimestampRange devono essere impostati.

Tipo: oggetto [DASHFragmentSelector](#)

Campo obbligatorio: no

[DisplayFragmentNumber](#)

I frammenti vengono identificati nel file manifesto in base al numero di sequenza nella sessione. Se DisplayFragmentNumber è impostato suALWAYS, il numero di frammento di Kinesis Video Streams viene aggiunto a ciascun elemento S nel file manifest con il nome dell'attributo «kvs:fn». Questi numeri di frammento possono essere utilizzati per la registrazione o per l'uso con altre API (ad esempio and). GetMedia GetMediaForFragmentList È necessario un lettore multimediale MPEG-DASH personalizzato per sfruttare questi attributi personalizzati.

Il valore predefinito è NEVER.

–Tipo: stringa

Valori validi: ALWAYS | NEVER

Campo obbligatorio: no

[DisplayFragmentTimestamp](#)

Secondo le specifiche MPEG-DASH, l'ora da parete dei frammenti nel file manifesto può essere derivata utilizzando gli attributi presenti nel manifesto stesso. Tuttavia, in genere, i lettori multimediali compatibili con MPEG-DASH non gestiscono correttamente le lacune nella timeline multimediale. Kinesis Video Streams regola la timeline multimediale nel file manifesto per consentire la riproduzione di contenuti multimediali con discontinuità. Pertanto, l'ora dell'orologio da parete derivata dal file manifesto potrebbe essere imprecisa. Se DisplayFragmentTimestamp è impostato suALWAYS, il timestamp preciso del frammento viene aggiunto a ogni elemento S nel file manifesto con il nome dell'attributo «kvs:ts». È necessario un lettore multimediale MPEG-DASH personalizzato per sfruttare questo attributo personalizzato.

Il valore predefinito è NEVER. In caso [DASHFragmentSelector](#) affermativoSERVER_TIMESTAMP, i timestamp saranno i timestamp di inizio del server. Allo stesso modo, quando

[DASHFragmentSelector](#) è così PRODUCER_TIMESTAMP, i timestamp saranno i timestamp di inizio del produttore.

▪Tipo: stringa

Valori validi: ALWAYS | NEVER

Campo obbligatorio: no

[Expires](#)

Il tempo in secondi che manca alla scadenza della sessione richiesta. Questo valore può essere compreso tra 300 (5 minuti) e 43200 (12 ore).

Quando una sessione scade, non è GetMP4MediaFragment possibile effettuare nuove chiamate a GetDashManifestGetMP4InitFragment, o per quella sessione.

Il valore predefinito è 300 (5 minuti).

Tipo: integer

Intervallo valido: valore minimo di 300. valore massimo pari a 43200.

Campo obbligatorio: no

[MaxManifestFragmentResults](#)

Il numero massimo di frammenti restituiti nel manifesto MPEG-DASH.

Quando PlaybackMode è LIVE, vengono restituiti i frammenti più recenti fino a questo valore. Quando PlaybackMode è ON_DEMAND, vengono restituiti i frammenti più vecchi, fino a questo numero massimo.

Quando il numero di frammenti disponibili in un manifesto MPEG-DASH live è elevato, i lettori video spesso memorizzano il contenuto nel buffer prima di iniziare la riproduzione. L'aumento della dimensione del buffer aumenta la latenza di riproduzione, ma diminuisce la probabilità che si verifichi il rebuffering durante la riproduzione. Si consiglia che un manifesto MPEG-DASH live contenga un minimo di 3 frammenti e un massimo di 10 frammenti.

L'impostazione predefinita è 5 frammenti se è LIVE o LIVE_REPLAY e 1.000 se lo PlaybackMode è. PlaybackMode ON_DEMAND

Il valore massimo di 1.000 frammenti corrisponde a più di 16 minuti di video in stream con frammenti di 1 secondo e più di 2 ore e mezza di video in stream con frammenti di 10 secondi.

Tipo: long

Intervallo valido: valore minimo di 1. Valore massimo di 5000.

Campo obbligatorio: no

PlaybackMode

Sia che si tratti di recuperare dati live, replay dal vivo o archiviati su richiesta.

Le caratteristiche dei tre tipi di sessioni includono quanto segue:

- **LIVE:** Per sessioni di questo tipo, il manifesto MPEG-DASH viene continuamente aggiornato con i frammenti più recenti non appena diventano disponibili. Si consiglia al lettore multimediale di recuperare un nuovo manifesto a intervalli di un secondo. Quando questo tipo di sessione viene riprodotta in un lettore multimediale, l'interfaccia utente in genere visualizza una notifica «live», senza alcun controllo di scorrimento per scegliere la posizione da visualizzare nella finestra di riproduzione.

Note

In LIVE modalità, i frammenti più recenti disponibili sono inclusi in un manifesto MPEG-DASH, anche se c'è uno spazio tra i frammenti (ovvero se ne manca uno). Un intervallo come questo potrebbe causare l'arresto di un lettore multimediale o un salto nella riproduzione. In questa modalità, i frammenti non vengono aggiunti al manifesto MPEG-DASH se sono più vecchi del frammento più recente della playlist. Se il frammento mancante diventa disponibile dopo l'aggiunta di un frammento successivo al manifesto, il frammento più vecchio non viene aggiunto e lo spazio vuoto non viene colmato.

- **LIVE_REPLAY:** Per le sessioni di questo tipo, il manifesto MPEG-DASH viene aggiornato in modo analogo a come viene aggiornato per la LIVE modalità, tranne per il fatto che inizia includendo i frammenti a partire da una determinata ora di inizio. Invece di aggiungere i frammenti man mano che vengono ingeriti, i frammenti vengono aggiunti man mano che trascorre la durata del frammento successivo. Ad esempio, se i frammenti della sessione durano due secondi, un nuovo frammento viene aggiunto al manifesto ogni due secondi. Questa modalità è utile per poter avviare la riproduzione dal momento in cui viene rilevato un evento e continuare lo streaming live di contenuti multimediali che non sono ancora stati acquisiti al momento della creazione della sessione. Questa modalità è utile anche per lo streaming di contenuti multimediali precedentemente archiviati senza essere limitata dal limite di 1.000 frammenti previsto dalla modalità. ON_DEMAND

- **ON_DEMAND**: Per le sessioni di questo tipo, il manifesto MPEG-DASH contiene tutti i frammenti della sessione, fino al numero specificato in `MaxManifestFragmentResults`. Il manifesto deve essere recuperato solo una volta per ogni sessione. Quando questo tipo di sessione viene riprodotta in un lettore multimediale, l'interfaccia utente in genere visualizza un controllo a scorrimento per scegliere la posizione da visualizzare nella finestra di riproduzione.

In tutte le modalità di riproduzione, se si `FragmentSelectorTypePRODUCER_TIMESTAMP`, e se sono presenti più frammenti con lo stesso timestamp di inizio, il frammento con il numero di frammento maggiore (ovvero il frammento più recente) viene incluso nel manifesto MPEG-DASH. Gli altri frammenti non sono inclusi. I frammenti con timestamp diversi ma con durate sovrapposte sono ancora inclusi nel manifesto MPEG-DASH. Ciò può portare a un comportamento imprevisto nel lettore multimediale.

Il valore predefinito è `LIVE`.

•Tipo: stringa

Valori validi: `LIVE` | `LIVE_REPLAY` | `ON_DEMAND`

Campo obbligatorio: no

[StreamARN](#)

L'Amazon Resource Name (ARN) dello stream per il quale recuperare l'URL del manifesto MPEG-DASH.

È necessario specificare il o il `StreamName` `StreamARN`

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

[StreamName](#)

Il nome dello stream per il quale recuperare l'URL del manifesto MPEG-DASH.

È necessario specificare il o il `StreamName` `StreamARN`

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "DASHStreamingSessionURL": "string"
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

DASHStreamingSessionURL

L'URL (contenente il token di sessione) che un lettore multimediale può utilizzare per recuperare il manifesto MPEG-DASH.

▪Tipo: stringa

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché hai superato un limite. Prova a effettuare la chiamata più tardi. Per informazioni sui limiti, consulta [Kinesis Video Streams Limits](#).

Codice di stato HTTP: 400

InvalidArgumentException

Un parametro specificato supera le restrizioni, non è supportato o non può essere utilizzato.

Codice di stato HTTP: 400

InvalidCodecPrivateDataException

I dati privati del codec in almeno una delle tracce del flusso video non sono validi per questa operazione.

Codice di stato HTTP: 400

MissingCodecPrivateDataException

Non è stato trovato alcun dato privato del codec in almeno una delle tracce del flusso video.

Codice di stato HTTP: 400

NoDataRetentionException

GetImages è stato richiesto per uno stream che non conserva dati (ovvero ha un valore `DataRetentionInHours` pari a 0).

Codice di stato HTTP: 400

NotAuthorizedException

Codice di stato: 403, il chiamante non è autorizzato a eseguire un'operazione sullo stream specificato o il token è scaduto.

Codice di stato HTTP: 401

ResourceNotFoundException

GetImages genererà questo errore quando Kinesis Video Streams non riesce a trovare lo stream specificato.

GetHLSStreamingSessionURL e GetDASHStreamingSessionURL genera questo errore se LIVE_REPLAY viene richiesta una sessione con un `PlaybackMode` di `ON_DEMAND` o per uno stream che non contiene frammenti nell'intervallo di tempo richiesto, o se LIVE viene richiesta una sessione con un `PlaybackMode` di `ON_DEMAND` per uno stream che non contiene frammenti negli ultimi 30 secondi.

Codice di stato HTTP: 404

UnsupportedStreamMediaTypeException

Il tipo di supporto (ad esempio, video h.264 o h.265 o audio AAC o G.711) non può essere determinato dagli ID dei codec delle tracce nel primo frammento di una sessione di riproduzione.

L'ID del codec per la traccia 1 dovrebbe essere V_MPEG/ISO/AVC e, facoltativamente, l'ID del codec per la traccia 2 dovrebbe essere. A_AAC

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli SDK specifici della lingua, consulta quanto segue AWS :

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

GetHLSStreamingSessionURL

Servizio: Amazon Kinesis Video Streams Archived Media

Recupera un URL HTTP Live Streaming (HLS) per lo stream. È quindi possibile aprire l'URL in un browser o lettore multimediale per visualizzare i contenuti dello streaming.

StreamName e i parametri che i StreamARN parametri sono facoltativi, ma è necessario specificare StreamName o il StreamARN quando si richiama questa operazione API.

Uno stream video di Amazon Kinesis presenta i seguenti requisiti per la fornitura di dati tramite HLS:

- Requisiti per la [riproduzione dei brani video](#).
- La conservazione dei dati deve essere superiore a 0.
- [La traccia video di ogni frammento deve contenere dati privati del codec in formato Advanced Video Coding \(AVC\) per H.264 o HEVC per il formato H.265 \(specifica MPEG-4 ISO/IEC 14496-15\)](#). Per informazioni sull'adattamento dei dati di flusso a un dato formato, vedi i [contrassegni di adattamento NAL](#).
- [La traccia audio \(se presente\) di ogni frammento deve contenere dati privati del codec in formato AAC \(specifica AAC ISO/IEC 13818-7\)](#).

Le sessioni HLS di Kinesis Video Streams contengono frammenti nel formato frammentato MPEG-4 (chiamato anche fMP4 o CMAF) o nel formato MPEG-2 (chiamato anche blocchi TS, supportato anche dalla specifica HLS). Per ulteriori informazioni sui tipi [di frammenti](#) HLS, consulta la specifica HLS.

La procedura seguente mostra come utilizzare HLS con Kinesis Video Streams:

1. Chiama l'GetDataEndpointAPI per ottenere un endpoint. Quindi invia le GetHLSStreamingSessionURL richieste a questo endpoint utilizzando il parametro [--endpoint-url](#).
2. Recupera l'URL HLS utilizzando. GetHLSStreamingSessionURL Kinesis Video Streams crea una sessione di streaming HLS da utilizzare per accedere ai contenuti di uno stream utilizzando il protocollo HLS. GetHLSStreamingSessionURL restituisce un URL autenticato (che include un token di sessione crittografato) per la playlist principale HLS della sessione (la risorsa principale necessaria per lo streaming con HLS).

 Note

Non condividere o archiviare questo token dove un'entità non autorizzata potrebbe accedervi. Il token fornisce l'accesso al contenuto dello stream. Proteggi il token con le stesse misure che utilizzeresti con AWS le tue credenziali.

I file multimediali resi disponibili tramite la playlist sono costituiti solo dallo stream, dall'intervallo di tempo e dal formato richiesti. Non sono disponibili altri dati multimediali (come frame esterni alla finestra richiesta o bitrate alternativi).

3. Fornisci l'URL (contenente il token di sessione crittografato) per la playlist principale HLS a un lettore multimediale che supporti il protocollo HLS. Kinesis Video Streams rende disponibili la playlist multimediale HLS, il frammento di inizializzazione e i frammenti multimediali tramite l'URL della playlist principale. Il frammento di inizializzazione contiene i dati privati del codec per lo streaming e altri dati necessari per configurare il decodificatore e il renderer video o audio. I frammenti multimediali contengono fotogrammi video con codifica H.264 o campioni audio con codifica AAC.
4. Il lettore multimediale riceve l'URL autenticato e richiede lo streaming di metadati e dati multimediali normalmente. Quando il lettore multimediale richiede dati, richiama le seguenti azioni:
 - `getTlsMasterPlaylist`: recupera una playlist principale HLS, che contiene un URL per l'azione `GetHLSMediaPlaylist` per ogni traccia e metadati aggiuntivi per il lettore multimediale, tra cui bitrate e risoluzione stimati.
 - `getTlsMediaPlaylist`: recupera una playlist multimediale HLS, che contiene un URL per accedere al frammento di inizializzazione MP4 con l'azione `GetMP4InitFragment` e gli URL per accedere ai frammenti multimediali MP4 con le azioni `GetMP4MediaFragment`. La playlist multimediale HLS contiene anche metadati sullo stream di cui il lettore ha bisogno per riprodurla, ad esempio se è `ON_DEMAND`. La playlist multimediale HLS è in genere statica per sessioni con un `di`. `PlaybackType ON_DEMAND` La playlist multimediale HLS viene continuamente aggiornata con nuovi frammenti per le sessioni con un `di`. `PlaybackType LIVE` Esiste una playlist multimediale HLS distinta per la traccia video e la traccia audio (se applicabile) che contiene gli URL multimediali MP4 per la traccia specifica.
 - `GetMP4InitFragment`: recupera il frammento di inizializzazione MP4. Il lettore multimediale in genere carica il frammento di inizializzazione prima di caricare qualsiasi frammento multimediale. Questo frammento contiene gli atomi "fyp" e "moov" MP4 e gli atomi secondari necessari per inizializzare il decoder del lettore multimediale.

Il frammento di inizializzazione non corrisponde a un frammento in uno stream video Kinesis. Contiene solo i dati privati del codec per lo stream e la rispettiva traccia, necessari al lettore multimediale per decodificare i frame multimediali.

- `GetMP4MediaFragment`: recupera i frammenti multimediali MP4. Questi frammenti contengono gli atomi "" e "moofmdat" MP4 e i relativi atomi secondari, contenenti i frame multimediali del frammento codificato e i relativi timestamp.

Note

I dati privati del codec (CPD) contenuti in ogni frammento contengono informazioni di inizializzazione specifiche del codec, come la frequenza dei fotogrammi, la risoluzione e il profilo di codifica, necessarie per decodificare correttamente il frammento. Sia per TS che per MP4, le modifiche CPD sono supportate durante una sessione di streaming. Pertanto, i frammenti di una sessione possono contenere informazioni diverse nel CPD senza interrompere la riproduzione. Per ogni sessione di streaming, sono consentite solo 500 modifiche CPD.

Important

Le modifiche alla traccia non sono supportate. Le tracce devono rimanere coerenti su tutti i file multimediali interrogati. Lo streaming fallirà se i frammenti dello stream passano da un formato contenente solo video ad avere sia audio che video, o se una traccia audio AAC viene sostituita da una traccia audio A-Law.

I dati recuperati con questa azione sono fatturabili. Per informazioni, consulta [Prezzi di](#).

- `getsFragment`: recupera i frammenti MPEG TS contenenti sia i dati di inizializzazione che i dati multimediali per tutte le tracce dello stream.

Note

In caso `affermativoMPEG_TS`, questa API `ContainerFormat` viene utilizzata al posto di `e` per recuperare i contenuti multimediali in streaming. `GetMP4InitFragment`
`GetMP4MediaFragment`

I dati recuperati con questa azione sono fatturabili. Per ulteriori informazioni, consulta i prezzi di [Kinesis Video Streams](#).

L'URL di una sessione di streaming non deve essere condiviso tra i giocatori. Il servizio potrebbe limitare una sessione se è condivisa da più lettori multimediali. Per i limiti di connessione, consulta [Kinesis Video Streams Limits](#).

Puoi monitorare la quantità di dati consumata dal lettore multimediale monitorando la CloudWatch metrica di `GetMP4MediaFragment.OutgoingBytes` Amazon. Per informazioni sull'utilizzo per CloudWatch monitorare Kinesis Video Streams, [consulta Monitoring Kinesis Video Streams](#). [Per informazioni sui prezzi, consulta Prezzi e prezzi di Amazon Kinesis Video Streams AWS](#) . Si applicano i costi sia per le sessioni HLS che per i dati in uscita AWS .

Guarda gli esempi di riproduzione video nella guida alla documentazione: e. [Usa AWS CLI per recuperare l'URL di una sessione di streaming HLS Esempio: usa HLS in HTML e JavaScript](#)

Per ulteriori informazioni su HLS, consulta [HTTP Live Streaming sul sito Apple Developer](#).

Important

Se viene generato un errore dopo aver richiamato un'API multimediale archiviata di Kinesis Video Streams, oltre al codice di stato HTTP e al corpo della risposta, include le seguenti informazioni:

- `x-amz-ErrorType` intestazione HTTP: contiene un tipo di errore più specifico oltre a quello fornito dal codice di stato HTTP.
- `x-amz-RequestId` intestazione HTTP: se desideri segnalare un problema AWS, il team di supporto può diagnosticare meglio il problema se gli viene fornito il Request Id.

Sia il codice di stato HTTP che l' `ErrorType` intestazione possono essere utilizzati per prendere decisioni programmatiche sulla possibilità di correggere gli errori e in quali condizioni, oltre a fornire informazioni sulle azioni che il programmatore client potrebbe dover intraprendere per riprovare con successo.

[Per ulteriori informazioni, consulta la sezione Errori nella parte inferiore di questo argomento, oltre a Errori comuni.](#)

Sintassi della richiesta

```
POST /getHLSStreamingSessionURL HTTP/1.1
Content-type: application/json

{
  "ContainerFormat": "string",
  "DiscontinuityMode": "string",
  "DisplayFragmentTimestamp": "string",
  "Expires": number,
  "HLSFragmentSelector": {
    "FragmentSelectorType": "string",
    "TimestampRange": {
      "EndTimestamp": number,
      "StartTimestamp": number
    }
  },
  "MaxMediaPlaylistFragmentResults": number,
  "PlaybackMode": "string",
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ContainerFormat

Specifica il formato da utilizzare per impacchettare il supporto. Specificando il formato del FRAGMENTED_MP4 contenitore, il file multimediale viene confezionato in frammenti MP4 (fMP4 o CMAF). Questa è la confezione consigliata perché le spese di imballaggio sono minime. L'altra opzione di formato del contenitore è MPEG_TS. HLS supporta i blocchi MPEG TS sin dal suo rilascio e a volte è l'unico pacchetto supportato sui lettori HLS meno recenti. MPEG TS ha in genere un sovraccarico di imballaggio del 5-25%. Ciò significa che MPEG TS richiede in genere il 5-25 per cento in più di larghezza di banda e costi rispetto a fMP4.

Il valore predefinito è FRAGMENTED_MP4.

▪Tipo: stringa

Valori validi: FRAGMENTED_MP4 | MPEG_TS

Campo obbligatorio: no

DiscontinuityMode

Specifica quando i flag che indicano le discontinuità tra i frammenti vengono aggiunti alle playlist multimediali.

I lettori multimediali in genere creano una sequenza temporale dei contenuti multimediali da riprodurre, in base ai timestamp di ciascun frammento. Ciò significa che se c'è una sovrapposizione o uno spazio tra i frammenti (come in genere se [HLSFragmentSelector](#) è impostata su `SERVER_TIMESTAMP`), la timeline del lettore multimediale presenterà anche piccoli spazi tra i frammenti in alcuni punti e sovrascriverà i fotogrammi in altri punti. Le lacune nella timeline del lettore multimediale possono causare il blocco della riproduzione e le sovrapposizioni possono causare nervosismo nella riproduzione. Quando sono presenti segnali di discontinuità tra i frammenti, si prevede che il lettore multimediale reimposti la timeline, in modo che il frammento successivo venga riprodotto immediatamente dopo il frammento precedente.

Sono supportate le seguenti modalità:

- **ALWAYS**: un indicatore di discontinuità viene inserito tra ogni frammento della playlist multimediale HLS. Si consiglia di utilizzare un valore pari a ALWAYS se i timestamp del frammento non sono accurati.
- **NEVER**: nessun indicatore di discontinuità viene posizionato da nessuna parte. Si consiglia di utilizzare un valore di per NEVER garantire che la timeline del lettore multimediale corrisponda con la massima precisione ai timestamp del produttore.
- **ON_DISCONTINUITY**: un marker di discontinuità viene posizionato tra i frammenti che presentano uno spazio o una sovrapposizione di oltre 50 millisecondi. Per la maggior parte degli scenari di riproduzione, si consiglia di utilizzare un valore pari a in ON_DISCONTINUITY modo che la timeline del lettore multimediale venga reimpostata solo in caso di problemi significativi con la timeline multimediale (ad esempio un frammento mancante).

L'impostazione predefinita è ALWAYS quando [HLSFragmentSelector](#) è impostata su `SERVER_TIMESTAMP`, e NEVER quando è impostata su `PRODUCER_TIMESTAMP`

▪Tipo: stringa

Valori validi: ALWAYS | NEVER | ON_DISCONTINUITY

Campo obbligatorio: no

[DisplayFragmentTimestamp](#)

Specifica quando i timestamp di inizio del frammento devono essere inclusi nella playlist multimediale HLS. In genere, i lettori multimediali segnalano la posizione della testina di riproduzione come un orario relativo all'inizio del primo frammento della sessione di riproduzione. Tuttavia, quando i timestamp di inizio sono inclusi nella playlist multimediale HLS, alcuni lettori multimediali potrebbero riportare la testina di riproduzione corrente come orario assoluto in base ai timestamp del frammento. Questo può essere utile per creare un'esperienza di riproduzione che mostri agli spettatori l'ora dell'orologio da parete del file multimediale.

Il valore predefinito è NEVER. In caso [HLSFragmentSelector](#) affermativoSERVER_TIMESTAMP, i timestamp saranno i timestamp di inizio del server. Allo stesso modo, quando [HLSFragmentSelector](#) è cosìPRODUCER_TIMESTAMP, i timestamp saranno i timestamp di inizio del produttore.

▪Tipo: stringa

Valori validi: ALWAYS | NEVER

Campo obbligatorio: no

[Expires](#)

Il tempo in secondi che manca alla scadenza della sessione richiesta. Questo valore può essere compreso tra 300 (5 minuti) e 43200 (12 ore).

Quando una sessione scade, non è GetTSFragment possibile effettuare nuove chiamate a GetHLSMasterPlaylist, GetHLSMediaPlaylist, GetMP4InitFragment, GetMP4MediaFragment,, o per quella sessione.

Il valore predefinito è 300 (5 minuti).

Tipo: integer

Intervallo valido: valore minimo di 300. valore massimo pari a 43200.

Campo obbligatorio: no

[HLSFragmentSelector](#)

L'intervallo di tempo del frammento richiesto e l'origine dei timestamp.

Questo parametro è obbligatorio se PlaybackMode è o. ON_DEMAND LIVE_REPLAY
Questo parametro è facoltativo se lo PlaybackMode è LIVE. Se PlaybackMode siLIVE, FragmentSelectorType può essere impostato, ma non TimestampRange deve essere impostato. Se PlaybackMode è ON_DEMAND oLIVE_REPLAY, entrambi FragmentSelectorType TimestampRange devono essere impostati.

Tipo: oggetto [HLSFragmentSelector](#)

Campo obbligatorio: no

[MaxMediaPlaylistFragmentResults](#)

Il numero massimo di frammenti restituiti nelle playlist multimediali HLS.

Quando PlaybackMode èLIVE, vengono restituiti i frammenti più recenti fino a questo valore. Quando PlaybackMode èON_DEMAND, vengono restituiti i frammenti più vecchi, fino a questo numero massimo.

Quando c'è un numero maggiore di frammenti disponibili in una playlist multimediale HLS live, i lettori video spesso memorizzano il contenuto nel buffer prima di iniziare la riproduzione. L'aumento della dimensione del buffer aumenta la latenza di riproduzione, ma diminuisce la probabilità che si verifichi il rebuffering durante la riproduzione. Consigliamo che una playlist multimediale HLS live contenga un minimo di 3 frammenti e un massimo di 10 frammenti.

L'impostazione predefinita è 5 frammenti se PlaybackMode è LIVE o e 1.000 se LIVE_REPLAY lo è. PlaybackMode ON_DEMAND

Il valore massimo di 5.000 frammenti corrisponde a più di 80 minuti di video in stream con frammenti di 1 secondo e a più di 13 ore di video in stream con frammenti di 10 secondi.

Tipo: long

Intervallo valido: valore minimo di 1. Valore massimo di 5000.

Campo obbligatorio: no

[PlaybackMode](#)

Sia che si tratti di recuperare dati live, replay dal vivo o archiviati su richiesta.

Le caratteristiche dei tre tipi di sessioni includono quanto segue:

- **LIVE:** Per sessioni di questo tipo, la playlist multimediale HLS viene continuamente aggiornata con i frammenti più recenti non appena diventano disponibili. Si consiglia al lettore multimediale di recuperare una nuova playlist a intervalli di un secondo. Quando questo tipo di sessione viene riprodotta in un lettore multimediale, l'interfaccia utente in genere visualizza una notifica «dal vivo», senza alcun controllo di scorrimento per la scelta della posizione nella finestra di riproduzione da visualizzare.

 Note

In LIVE modalità, i frammenti più recenti disponibili vengono inclusi in una playlist multimediale HLS, anche se c'è uno spazio tra i frammenti (ovvero se manca un frammento). Un intervallo come questo potrebbe causare l'arresto di un lettore multimediale o causare un salto nella riproduzione. In questa modalità, i frammenti non vengono aggiunti alla playlist multimediale HLS se sono più vecchi del frammento più recente della playlist. Se il frammento mancante diventa disponibile dopo l'aggiunta di un frammento successivo alla playlist, il frammento più vecchio non viene aggiunto e lo spazio vuoto non viene colmato.

- **LIVE_REPLAY:** Per le sessioni di questo tipo, la playlist multimediale HLS viene aggiornata in modo simile a come viene aggiornata per la LIVE modalità, tranne per il fatto che inizia includendo i frammenti di una determinata ora di inizio. Invece di aggiungere i frammenti man mano che vengono ingeriti, i frammenti vengono aggiunti man mano che trascorre la durata del frammento successivo. Ad esempio, se i frammenti della sessione durano due secondi, viene aggiunto un nuovo frammento alla playlist multimediale ogni due secondi. Questa modalità è utile per poter avviare la riproduzione dal momento in cui viene rilevato un evento e continuare lo streaming live di contenuti multimediali che non sono ancora stati acquisiti al momento della creazione della sessione. Questa modalità è utile anche per lo streaming di contenuti multimediali precedentemente archiviati senza essere limitata dal limite di 1.000 frammenti previsto dalla modalità. ON_DEMAND
- **ON_DEMAND:** Per sessioni di questo tipo, la playlist multimediale HLS contiene tutti i frammenti della sessione, fino al numero specificato in `MaxMediaPlaylistFragmentResults`. La playlist deve essere recuperata solo una volta per ogni sessione. Quando questo tipo di sessione viene riprodotta in un lettore multimediale, l'interfaccia utente in genere visualizza un comando a scorrimento per scegliere la posizione da visualizzare nella finestra di riproduzione.

In tutte le modalità di riproduzione, se si `FragmentSelectorTypePRODUCER_TIMESTAMP`, e se sono presenti più frammenti con lo stesso timestamp di inizio, il frammento con il

numero di frammenti più elevato (ovvero il frammento più recente) viene incluso nella playlist multimediale HLS. Gli altri frammenti non sono inclusi. I frammenti con timestamp diversi ma con durate sovrapposte sono ancora inclusi nella playlist multimediale HLS. Ciò può portare a un comportamento imprevisto nel lettore multimediale.

Il valore predefinito è LIVE.

▀Tipo: stringa

Valori validi: LIVE | LIVE_REPLAY | ON_DEMAND

Campo obbligatorio: no

StreamARN

L'Amazon Resource Name (ARN) dello stream per il quale recuperare l'URL della playlist principale HLS.

Devi specificare il o il. StreamName StreamARN

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+

Campo obbligatorio: no

StreamName

Il nome dello stream per il quale recuperare l'URL della playlist principale HLS.

È necessario specificare il StreamName o il. StreamARN

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "HLSStreamingSessionURL": "string"
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

[HLSStreamingSessionURL](#)

L'URL (contenente il token di sessione) che un lettore multimediale può utilizzare per recuperare la playlist principale HLS.

▪Tipo: stringa

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché hai superato un limite. Prova a effettuare la chiamata più tardi. Per informazioni sui limiti, consulta [Kinesis Video Streams Limits](#).

Codice di stato HTTP: 400

InvalidArgumentException

Un parametro specificato supera le restrizioni, non è supportato o non può essere utilizzato.

Codice di stato HTTP: 400

InvalidCodecPrivateDataException

I dati privati del codec in almeno una delle tracce del flusso video non sono validi per questa operazione.

Codice di stato HTTP: 400

MissingCodecPrivateDataException

Non è stato trovato alcun dato privato del codec in almeno una delle tracce del flusso video.

Codice di stato HTTP: 400

NoDataRetentionException

GetImages è stato richiesto per uno stream che non conserva dati (ovvero ha un valore `DataRetentionInHours` pari a 0).

Codice di stato HTTP: 400

NotAuthorizedException

Codice di stato: 403, il chiamante non è autorizzato a eseguire un'operazione sullo stream specificato o il token è scaduto.

Codice di stato HTTP: 401

ResourceNotFoundException

GetImages genererà questo errore quando Kinesis Video Streams non riesce a trovare lo stream specificato.

GetHLSStreamingSessionURL e GetDASHStreamingSessionURL genera questo errore se LIVE_REPLAY viene richiesta una sessione con un `PlaybackMode` di `ON_DEMAND` o per uno stream che non contiene frammenti nell'intervallo di tempo richiesto, o se LIVE viene richiesta una sessione con un `PlaybackMode` di per uno stream che non contiene frammenti negli ultimi 30 secondi.

Codice di stato HTTP: 404

UnsupportedStreamMediaTypeException

Il tipo di supporto (ad esempio, video h.264 o h.265 o audio AAC o G.711) non può essere determinato dagli ID dei codec delle tracce nel primo frammento di una sessione di riproduzione. L'ID del codec per la traccia 1 dovrebbe essere `V_MPEG/ISO/AVC` e, facoltativamente, l'ID del codec per la traccia 2 dovrebbe essere `A_AAC`.

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli SDK specifici della lingua, consulta quanto segue AWS :

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

GetImages

Servizio: Amazon Kinesis Video Streams Archived Media

Recupera un elenco di immagini corrispondenti a ciascun timestamp per un determinato intervallo di tempo, intervallo di campionamento e configurazione del formato dell'immagine.

Note

Devi prima chiamare l'GetDataEndpointAPI per ottenere un endpoint. Quindi invia le GetImages richieste a questo endpoint utilizzando il parametro [--endpoint-url](#).

[Requisiti per la riproduzione dei brani video.](#)

Sintassi della richiesta

```
POST /getImages HTTP/1.1
Content-type: application/json

{
  "EndTimeStamp": number,
  "Format": "string",
  "FormatConfig": {
    "string" : "string"
  },
  "HeightPixels": number,
  "ImageSelectorType": "string",
  "MaxResults": number,
  "NextToken": "string",
  "SamplingInterval": number,
  "StartTimeStamp": number,
  "StreamARN": "string",
  "StreamName": "string",
  "WidthPixels": number
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

EndTimeStamp

Il timestamp di fine dell'intervallo di immagini da generare. Se l'intervallo di tempo compreso tra `StartTimeStamp` e `EndTimeStamp` supera i 300 secondi `StartTimeStamp`, riceverai un `IllegalArgumentException`

Tipo: `Timestamp`

Campo obbligatorio: sì

Format

Il formato che verrà utilizzato per codificare l'immagine.

▪Tipo: `stringa`

Valori validi: `JPEG` | `PNG`

Campo obbligatorio: sì

FormatConfig

L'elenco di una struttura di coppie chiave-valore che contiene parametri aggiuntivi che possono essere applicati quando viene generata l'immagine. La `FormatConfig` chiave è la `JPEGQuality`, che indica la chiave di qualità JPEG da utilizzare per generare l'immagine. Il `FormatConfig` valore accetta `int` da 1 a 100. Se il valore è 1, l'immagine verrà generata con una qualità inferiore e la compressione migliore. Se il valore è 100, l'immagine verrà generata con la migliore qualità e una compressione inferiore. Se non viene fornito alcun valore, il valore predefinito della `JPEGQuality` chiave verrà impostato su 80.

Tipo: mappatura stringa a stringa

Voci sulla mappa: numero massimo di 1 elemento.

Chiavi valide: `JPEGQuality`

Limiti di lunghezza del valore: lunghezza minima di 0. La lunghezza massima è 256 caratteri.

Modello di valore: `^[a-zA-Z_0-9]+`

Campo obbligatorio: no

HeightPixels

L'altezza dell'immagine di output utilizzata insieme al parametro. `WidthPixels` Quando vengono forniti entrambi `HeightPixels` e `WidthPixels` parametri, l'immagine verrà allungata per adattarsi alle proporzioni specificate. Se viene fornito solo il `HeightPixels` parametro, per calcolare il rapporto verranno utilizzate le `WidthPixels` proporzioni originali. Se non viene fornito alcun parametro, verrà restituita la dimensione originale dell'immagine.

Tipo: integer

Intervallo valido: valore minimo di 1. Valore massimo di 2160.

Campo obbligatorio: no

ImageSelectorType

L'origine dei timestamp del Server o del Produttore da utilizzare per generare le immagini.

▪Tipo: stringa

Valori validi: `PRODUCER_TIMESTAMP` | `SERVER_TIMESTAMP`

Campo obbligatorio: sì

MaxResults

Il numero massimo di immagini che devono essere restituite dall'API.

Note

Il limite predefinito è di 25 immagini per risposta API. Fornendo un valore `MaxResults` superiore a questo si otterrà una dimensione della pagina di 25. Eventuali risultati aggiuntivi verranno suddivisi in pagine.

Tipo: long

Intervallo valido: valore minimo di 1. valore massimo pari a 100.

Campo obbligatorio: no

NextToken

Un token che specifica da dove iniziare a impaginare il prossimo set di immagini. Questo è il risultato di una `GetImages:NextToken` risposta precedentemente troncata.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. Lunghezza massima di 4096.

Modello: `[a-zA-Z0-9+/]{0,2}`

Campo obbligatorio: no

SamplingInterval

L'intervallo di tempo in millisecondi (ms) in cui le immagini devono essere generate dallo stream. Il valore minimo che può essere fornito è 200 ms (5 immagini al secondo). Se l'intervallo del timestamp è inferiore all'intervallo di campionamento, l'immagine da `startTimestamp` verrà restituita, se disponibile.

Tipo: integer

Campo obbligatorio: sì

StartTimestamp

Il punto di partenza da cui devono essere generate le immagini. Affinché un'immagine possa essere restituita, questo `StartTimestamp` deve rientrare in un intervallo completo di timestamp.

Tipo: Timestamp

Campo obbligatorio: sì

StreamARN

L'Amazon Resource Name (ARN) dello stream da cui recuperare le immagini. È necessario specificare il `StreamName` o il `StreamARN`

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome dello stream da cui recuperare le immagini. È necessario specificare il `StreamName` o `StreamARN`.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

WidthPixels

La larghezza dell'immagine di output utilizzata insieme al `HeightPixels` parametro. Quando vengono forniti entrambi `WidthPixels` e `HeightPixels` parametri, l'immagine verrà allungata per adattarsi alle proporzioni specificate. Se viene fornito solo il `WidthPixels` parametro o se `HeightPixels` viene fornito solo il, `ValidationException` verrà generato un. Se non viene fornito nessun parametro, verrà restituita la dimensione dell'immagine originale dello stream.

Tipo: integer

Intervallo valido: valore minimo di 1. Valore massimo di 3840.

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "Images": [
    {
      "Error": "string",
      "ImageContent": "string",
      "TimeStamp": number
    }
  ],
  "NextToken": "string"
}
```

```
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

Images

L'elenco delle immagini generate dal flusso video. Se non è disponibile alcun supporto per il timestamp specificato, l'NO_MEDIAerrore verrà elencato nell'output. Se si verifica un errore durante la generazione dell'immagine, MEDIA_ERROR questo verrà elencato nell'output come causa dell'immagine mancante.

Tipo: matrice di oggetti [Image](#)

NextToken

Il token crittografato utilizzato nella richiesta per ottenere altre immagini.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. Lunghezza massima di 4096.

Modello: `[a-zA-Z0-9+/\]+= {0, 2}`

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché hai superato un limite. Prova a effettuare la chiamata più tardi. Per informazioni sui limiti, consulta [Kinesis Video Streams Limits](#).

Codice di stato HTTP: 400

InvalidArgumentException

Un parametro specificato supera le restrizioni, non è supportato o non può essere utilizzato.

Codice di stato HTTP: 400

NotAuthorizedException

Codice di stato: 403, il chiamante non è autorizzato a eseguire un'operazione sullo stream specificato o il token è scaduto.

Codice di stato HTTP: 401

ResourceNotFoundException

GetImagesgenererà questo errore quando Kinesis Video Streams non riesce a trovare lo stream specificato.

GetHLSStreamingSessionURL e GetDASHStreamingSessionURL genera questo errore se LIVE_REPLAY viene richiesta una sessione con un PlaybackMode of ON_DEMAND o per uno stream che non contiene frammenti nell'intervallo di tempo richiesto, o se LIVE viene richiesta una sessione con un PlaybackMode of per uno stream che non contiene frammenti negli ultimi 30 secondi.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

GetMediaForFragmentList

Servizio: Amazon Kinesis Video Streams Archived Media

Ottiene i file multimediali per un elenco di frammenti (specificato in base al numero di frammento) dai dati archiviati in un flusso video di Amazon Kinesis.

Note

Devi prima chiamare l'`GetDataEndpointAPI` per ottenere un endpoint. Quindi invia le `GetMediaForFragmentList` richieste a questo endpoint utilizzando il parametro `--endpoint-url`.

Per i limiti, consulta [Kinesis Video Streams Limits](#).

Important

Se viene generato un errore dopo aver richiamato un'API multimediale archiviata di Kinesis Video Streams, oltre al codice di stato HTTP e al corpo della risposta, include le seguenti informazioni:

- `x-amz-ErrorType` intestazione HTTP: contiene un tipo di errore più specifico oltre a quello fornito dal codice di stato HTTP.
- `x-amz-RequestId` intestazione HTTP: se desideri segnalare un problema AWS, il team di supporto può diagnosticare meglio il problema se gli viene fornito il Request Id.

Sia il codice di stato HTTP che l' `ErrorType` intestazione possono essere utilizzati per prendere decisioni programmatiche sulla possibilità di correggere gli errori e in quali condizioni, oltre a fornire informazioni sulle azioni che il programmatore client potrebbe dover intraprendere per riprovare con successo.

[Per ulteriori informazioni, consulta la sezione Errori nella parte inferiore di questo argomento, oltre a Errori comuni.](#)

Sintassi della richiesta

```
POST /getMediaForFragmentList HTTP/1.1
Content-type: application/json
```

```
{
  "Fragments": [ "string" ],
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

Fragments

Un elenco del numero di frammenti per i quali recuperare i file multimediali. Recuperate questi valori con. [ListFragments](#)

Tipo: matrice di stringhe

Membri dell'array: numero minimo di 1 elemento. Numero massimo di 1.000 elementi.

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: $^{\wedge}[\text{0-9}]^{\wedge}+$

Campo obbligatorio: sì

StreamARN

L'Amazon Resource Name (ARN) dello stream da cui recuperare i frammenti multimediali. Specificate questo parametro o il parametro. `StreamName`

-Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]^+`

Campo obbligatorio: no

StreamName

Il nome dello stream da cui recuperare i frammenti multimediali. Specificate questo parametro o il `StreamARN` parametro.

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-Type: ContentType

Payload
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

La risposta restituisce le seguenti intestazioni HTTP.

ContentType

Il tipo di contenuto del supporto richiesto.

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: `^[a-zA-Z0-9_\.\\-]+$`

La risposta restituisce quanto segue come corpo HTTP.

Payload

Il payload restituito da Kinesis Video Streams è una sequenza di blocchi del flusso specificato. Per informazioni sui blocchi, consulta [PutMedia](#). I blocchi che Kinesis Video Streams restituisce nella chiamata includono anche i seguenti tag `GetMediaForFragmentList` Matroska (MKV) aggiuntivi:

- `AWS_KINESISVIDEO_FRAGMENT_NUMBER` - Numero di frammento restituito nel blocco.
- `AWS_KINESISVIDEO_SERVER_SIDE_TIMESTAMP` - Timestamp sul lato server del frammento.
- `AWS_KINESISVIDEO_PRODUCER_SIDE_TIMESTAMP` - Timestamp - Timestamp sul lato produttore del frammento.

I seguenti tag verranno inclusi se si verifica un'eccezione:

- `AWS_KINESISVIDEO_FRAGMENT_NUMBER` - Il numero del frammento che ha generato l'eccezione.
- `AWS_KINESISVIDEO_EXCEPTION_ERROR_CODE` - Il codice intero dell'errore.
- `AWS_KINESISVIDEO_EXCEPTION_MESSAGE` - Una descrizione testuale dell'eccezione.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché hai superato un limite. Prova a effettuare la chiamata più tardi. Per informazioni sui limiti, consulta [Kinesis Video Streams Limits](#).

Codice di stato HTTP: 400

InvalidArgumentException

Un parametro specificato supera le restrizioni, non è supportato o non può essere utilizzato.

Codice di stato HTTP: 400

NotAuthorizedException

Codice di stato: 403, il chiamante non è autorizzato a eseguire un'operazione sullo stream specificato o il token è scaduto.

Codice di stato HTTP: 401

ResourceNotFoundException

GetImages genererà questo errore quando Kinesis Video Streams non riesce a trovare lo stream specificato.

GetHLSStreamingSessionURL e GetDASHStreamingSessionURL genera questo errore se LIVE_REPLAY viene richiesta una sessione con un PlaybackMode of ON_DEMAND o per uno stream che non contiene frammenti nell'intervallo di tempo richiesto, o se LIVE viene richiesta una sessione con un PlaybackMode of per uno stream che non contiene frammenti negli ultimi 30 secondi.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

ListFragments

Servizio: Amazon Kinesis Video Streams Archived Media

Restituisce un elenco di [Fragment](#) oggetti dal flusso e dall'intervallo di timestamp specificati all'interno dei dati archiviati.

L'elenco dei frammenti alla fine è coerente. Ciò significa che, anche se il produttore riceve una conferma che un frammento è persistente, il risultato potrebbe non essere restituito immediatamente da una richiesta a `ListFragments`. Tuttavia, i risultati sono generalmente disponibili in meno di un secondo.

Note

Devi prima chiamare `GetDataEndpointAPI` per ottenere un endpoint. Quindi invia le `ListFragments` richieste a questo endpoint utilizzando il parametro `--endpoint-url`.

Important

Se viene generato un errore dopo aver richiamato un'API multimediale archiviata di Kinesis Video Streams, oltre al codice di stato HTTP e al corpo della risposta, include le seguenti informazioni:

- `x-amz-ErrorType` intestazione HTTP: contiene un tipo di errore più specifico oltre a quello fornito dal codice di stato HTTP.
- `x-amz-RequestId` intestazione HTTP: se desideri segnalare un problema AWS, il team di supporto può diagnosticare meglio il problema se gli viene fornito il Request Id.

Sia il codice di stato HTTP che l' `ErrorType` intestazione possono essere utilizzati per prendere decisioni programmatiche sulla possibilità di correggere gli errori e in quali condizioni, oltre a fornire informazioni sulle azioni che il programmatore client potrebbe dover intraprendere per riprovare con successo.

[Per ulteriori informazioni, consulta la sezione Errori nella parte inferiore di questo argomento, nonché Errori comuni.](#)

Sintassi della richiesta

```
POST /listFragments HTTP/1.1
Content-type: application/json

{
  "FragmentSelector": {
    "FragmentSelectorType": "string",
    "TimestampRange": {
      "EndTimeStamp": number,
      "StartTimeStamp": number
    }
  },
  "MaxResults": number,
  "NextToken": "string",
  "StreamARN": "string",
  "StreamName": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

FragmentSelector

Descrive l'intervallo di timestamp e l'origine del timestamp per l'intervallo di frammenti da restituire.

Note

Questo è richiesto solo quando NextToken non viene passato nell'API.

Tipo: oggetto [FragmentSelector](#)

Campo obbligatorio: no

MaxResults

Il numero totale di frammenti da restituire. Se il numero totale di frammenti disponibili è superiore al valore specificato in `max-results`, nell'output `NextToken` viene fornito un [ListFragments](#): che è possibile utilizzare per riprendere l'impaginazione.

Il valore predefinito è 100.

Tipo: long

Intervallo valido: valore minimo di 1. Valore massimo pari a 1000.

Campo obbligatorio: no

NextToken

Token per specificare dove iniziare l'impaginazione. Questo è il [ListFragments: di una risposta NextToken troncata](#) in precedenza.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. Lunghezza massima di 4096.

Modello: `[a-zA-Z0-9+/]{0,2}`

Campo obbligatorio: no

StreamARN

L'Amazon Resource Name (ARN) dello stream da cui recuperare un elenco di frammenti. Specificare questo parametro o il parametro `StreamName`

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome dello stream da cui recuperare un elenco di frammenti. Specificate questo parametro o il `StreamARN` parametro.

•Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "Fragments": [
    {
      "FragmentLengthInMilliseconds": number,
      "FragmentNumber": "string",
      "FragmentSizeInBytes": number,
      "ProducerTimestamp": number,
      "ServerTimestamp": number
    }
  ],
  "NextToken": "string"
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

Fragments

Un elenco di [Fragment](#) oggetti archiviati dallo stream che soddisfano i criteri del selettore. I risultati non sono in ordine specifico, nemmeno tra le pagine.

Se nello stream non sono presenti frammenti che soddisfano i criteri del selettore, viene restituito un elenco vuoto.

Tipo: matrice di oggetti [Fragment](#)

NextToken

Se l'elenco restituito viene troncato, l'operazione restituisce questo token da utilizzare per recuperare la pagina successiva di risultati. Questo valore è null quando non ci sono altri risultati da restituire.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. Lunghezza massima di 4096.

Modello: `[a-zA-Z0-9+/\]+= {0, 2}`

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché hai superato un limite. Prova a effettuare la chiamata più tardi. Per informazioni sui limiti, consulta [Kinesis Video Streams Limits](#).

Codice di stato HTTP: 400

InvalidArgumentException

Un parametro specificato supera le restrizioni, non è supportato o non può essere utilizzato.

Codice di stato HTTP: 400

NotAuthorizedException

Codice di stato: 403, il chiamante non è autorizzato a eseguire un'operazione sullo stream specificato o il token è scaduto.

Codice di stato HTTP: 401

ResourceNotFoundException

GetImagesgenererà questo errore quando Kinesis Video Streams non riesce a trovare lo stream specificato.

GetHLSStreamingSessionURLe GetDASHStreamingSessionURL genera questo errore se LIVE_REPLAY viene richiesta una sessione con un PlaybackMode of ON_DEMAND o per uno stream che non contiene frammenti nell'intervallo di tempo richiesto, o se LIVE viene richiesta

una sessione con un PlaybackMode of per uno stream che non contiene frammenti negli ultimi 30 secondi.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

Flesso di video Amazon Kinesis

Le seguenti azioni sono supportate dai canali di segnalazione video di Amazon Kinesis:

- [GetIceServerConfig](#)
- [SendAlexaOfferToMaster](#)

GetIceServerConfig

Servizio: Amazon Kinesis Video Signaling Channels

Nota: prima di utilizzare questa API, è necessario chiamarla per richiedere l'endpoint HTTPS. `GetSignalingChannelEndpoint` Quindi specifichi l'endpoint e la regione nella richiesta `GetIceServerConfig` API.

Ottiene le informazioni sulla configurazione del server ICE (Interactive Connectivity Establishment), inclusi URI, nome utente e password che possono essere utilizzati per configurare la connessione WebRTC. Il componente ICE utilizza queste informazioni di configurazione per configurare la connessione WebRTC, inclusa l'autenticazione con il relay server Traversal Using Relays around NAT (TURN).

TURN è un protocollo utilizzato per migliorare la connettività delle applicazioni. peer-to-peer Fornendo un servizio di inoltro basato sul cloud, TURN garantisce la possibilità di stabilire una connessione anche quando uno o più peer non sono in grado di effettuare una connessione diretta. peer-to-peer Per ulteriori informazioni, consulta [Un'API REST per l'accesso ai servizi TURN](#).

Puoi richiamare questa API per stabilire un meccanismo di fallback nel caso in cui uno dei peer non sia in grado di stabilire una peer-to-peer connessione diretta tramite un canale di segnalazione. Devi specificare l'Amazon Resource Name (ARN) del tuo canale di segnalazione per richiamare questa API.

Sintassi della richiesta

```
POST /v1/get-ice-server-config HTTP/1.1
Content-type: application/json

{
  "ChannelARN": "string",
  "ClientId": "string",
  "Service": "string",
  "Username": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ChannelARN

L'ARN del canale di segnalazione da utilizzare per la peer-to-peer connessione tra peer configurati.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: sì

ClientId

Identificatore univoco per lo spettatore. Deve essere unico all'interno del canale di segnalazione.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Service

Specifica il servizio desiderato. Attualmente, TURN è l'unico valore valido.

▀Tipo: stringa

Valori validi: TURN

Campo obbligatorio: no

Username

Un ID utente opzionale da associare alle credenziali.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "IceServerList": [
    {
      "Password": "string",
      "Ttl": number,
      "Uris": [ "string" ],
      "Username": "string"
    }
  ]
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

[IceServerList](#)

L'elenco degli oggetti informativi del server ICE.

Tipo: matrice di oggetti [IceServer](#)

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

La tua richiesta è stata limitata perché hai superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

InvalidClientException

Il client specificato non è valido.

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire questa operazione.

Codice di stato HTTP: 401

ResourceNotFoundException

La risorsa specificata non è stata trovata.

Codice di stato HTTP: 404

SessionExpiredException

Se la sessione del client è scaduta. Una volta connesso il client, la sessione è valida per 45 minuti. Il client deve riconnettersi al canale per continuare a inviare/ricevere messaggi.

Codice di stato HTTP: 400

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli SDK specifici della lingua, consulta AWS quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

SendAlexaOfferToMaster

Servizio: Amazon Kinesis Video Signaling Channels

Note

Prima di utilizzare questa API, devi chiamarla per ottenere un endpoint. `GetSignalingChannelEndpoint` Devi quindi specificare l'endpoint e la regione nella tua richiesta `SendAlexaOfferToMaster` API.

Questa API ti consente di connettere dispositivi abilitati per WebRTC con dispositivi di visualizzazione Alexa. Quando viene richiamata, invia l'offerta Alexa Session Description Protocol (SDP) al peer principale. L'offerta viene fornita non appena il master viene collegato al canale di segnalazione specificato. Questa API restituisce la risposta SDP dal master connesso. Se il master non è collegato al canale di segnalazione, le richieste di riconsegna vengono effettuate fino alla scadenza del messaggio.

Sintassi della richiesta

```
POST /v1/send-alex-a-offer-to-master HTTP/1.1
Content-type: application/json

{
  "ChannelARN": "string",
  "MessagePayload": "string",
  "SenderClientId": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

ChannelARN

L'Amazon Resource Name (ARN) del canale di segnalazione tramite il quale Alexa e il master peer comunicano.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: sì

MessagePayload

L'SDP con codifica base64 offre contenuti.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. Lunghezza massima di 10000.

Modello: `[a-zA-Z0-9+/=]+`

Campo obbligatorio: sì

SenderClientId

L'identificatore univoco del client mittente.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: sì

Sintassi della risposta

```
HTTP/1.1 200
Content-type: application/json

{
  "Answer": "string"
}
```

Elementi di risposta

Se l'operazione riesce, il servizio restituisce una risposta HTTP 200.

I dati seguenti vengono restituiti in formato JSON mediante il servizio.

Answer

Il contenuto della risposta SDP con codifica in base 64.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. Lunghezza massima di 10000.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

ClientLimitExceededException

La tua richiesta è stata limitata perché hai superato il limite di chiamate consentite ai client. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

NotAuthorizedException

Il chiamante non è autorizzato a eseguire questa operazione.

Codice di stato HTTP: 401

ResourceNotFoundException

La risorsa specificata non è stata trovata.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per.NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

Amazon Kinesis Video Video Video Video Video Video Video Video WebRTC
Video Video Video Video Video Video Video Video

Le seguenti azioni sono supportate da Amazon Kinesis Video WebRTC Storage:

- [JoinStorageSession](#)

JoinStorageSession

Servizio: Amazon Kinesis Video WebRTC Storage

Note

Prima di utilizzare questa API, è necessario chiamare `GetSignalingChannelEndpointAPI` per richiedere l'endpoint WebRTC. È quindi necessario specificare l'endpoint e la regione nella richiesta API. `JoinStorageSession`

Partecipa alla sessione WebRTC audio unidirezionale e/o multidirezionale in corso come dispositivo di produzione video per un canale di ingresso. Se non esiste una sessione esistente per il canale, è necessario creare una nuova sessione di streaming e fornire l'Amazon Resource Name (ARN) del canale di segnalazione.

Attualmente, per `SINGLE_MASTER` questo tipo di dispositivo, un dispositivo di produzione video è in grado di importare contenuti audio e video in uno stream. Solo i dispositivi per la produzione di video possono partecipare alla sessione e registrare contenuti multimediali.

Important

Attualmente sono necessarie sia le tracce audio che quelle video per l'ingestione di WebRTC. Requisiti attuali:

- Traccia video: H.264
- Traccia audio: Opus

Il video acquisito risultante nel flusso video Kinesis avrà i seguenti parametri: video H.264 e audio AAC.

Una volta che un partecipante principale ha negoziato una connessione tramite WebRTC, la sessione multimediale acquisita verrà archiviata nel flusso video di Kinesis. Più spettatori possono quindi riprodurre contenuti multimediali in tempo reale tramite le nostre API di riproduzione.

Puoi anche utilizzare le HLS funzionalità esistenti di Kinesis Video Streams DASH come la riproduzione, la [GetImages](#) generazione di immagini tramite e altro ancora con i contenuti multimediali WebRTC incorporati.

Note

La consegna e le notifiche delle immagini S3 non sono attualmente supportate.

Note

Supponiamo che sia possibile associare un solo dispositivo client per la produzione di video a una sessione per il canale. Se più di un client si unisce alla sessione di un canale specifico come dispositivo di produzione video, la richiesta più recente del client ha la precedenza.

Informazioni aggiuntive

- Idempotente: questa API non è idempotente.
- Comportamento dei tentativi: viene conteggiata come una nuova chiamata API.
- Chiamate simultanee: le chiamate simultanee sono consentite. Un'offerta viene inviata una volta per ogni chiamata.

Sintassi della richiesta

```
POST /joinStorageSession HTTP/1.1
Content-type: application/json
```

```
{
  "channelArn": "string"
}
```

Parametri della richiesta URI:

La richiesta non utilizza parametri URI.

Corpo della richiesta

La richiesta accetta i seguenti dati in formato JSON.

channelArn

Il nome della risorsa Amazon (ARN) del canale di segnalazione.

Tipo: stringa

Modello: `^arn:(aws[a-zA-Z-]*):kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+$`

Campo obbligatorio: sì

Sintassi della risposta

```
HTTP/1.1 200
```

Elementi di risposta

Se l'operazione riesce, il servizio invia una risposta HTTP 200 con un corpo HTTP vuoto.

Errori

Per informazioni sugli errori comuni a tutte le operazioni, consultare [Errori comuni](#).

AccessDeniedException

Non disponi delle autorizzazioni necessarie per eseguire questa operazione.

Codice di stato HTTP: 403

ClientLimitExceededException

Kinesis Video Streams ha limitato la richiesta perché è stato superato il limite di chiamate client consentite. Prova a effettuare la chiamata più tardi.

Codice di stato HTTP: 400

InvalidArgumentException

Il valore di questo parametro di input non è valido.

Codice di stato HTTP: 400

ResourceNotFoundException

La risorsa specificata non è stata trovata.

Codice di stato HTTP: 404

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [Interfaccia a riga di comando AWS](#)
- [AWS SDK per .NET](#)
- [AWS SDK per C++](#)
- [AWS SDK per Go v2](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per V3 JavaScript](#)
- [AWS SDK per PHP V3](#)
- [AWS SDK per Python](#)
- [AWS SDK per Ruby V3](#)

Tipi di dati

I seguenti tipi di dati sono supportati da Amazon Kinesis Video Streams:

- [ChannelInfo](#)
- [ChannelNameCondition](#)
- [DeletionConfig](#)
- [EdgeAgentStatus](#)
- [EdgeConfig](#)
- [ImageGenerationConfiguration](#)
- [ImageGenerationDestinationConfig](#)
- [LastRecorderStatus](#)
- [LastUploaderStatus](#)
- [ListEdgeAgentConfigurationsEdgeConfig](#)
- [LocalSizeConfig](#)
- [MappedResourceConfigurationListItem](#)
- [MediaSourceConfig](#)
- [MediaStorageConfiguration](#)

- [NotificationConfiguration](#)
- [NotificationDestinationConfig](#)
- [RecorderConfig](#)
- [ResourceEndpointListItem](#)
- [ScheduleConfig](#)
- [SingleMasterChannelEndpointConfiguration](#)
- [SingleMasterConfiguration](#)
- [StreamInfo](#)
- [StreamNameCondition](#)
- [Tag](#)
- [UploaderConfig](#)

I seguenti tipi di dati sono supportati da Amazon Kinesis Video Streams Media:

- [StartSelector](#)

I seguenti tipi di dati sono supportati da Amazon Kinesis Video Streams Archived Media:

- [ClipFragmentSelector](#)
- [ClipTimestampRange](#)
- [DASHFragmentSelector](#)
- [DASHTimestampRange](#)
- [Fragment](#)
- [FragmentSelector](#)
- [HLSFragmentSelector](#)
- [HLSTimestampRange](#)
- [Image](#)
- [TimestampRange](#)

I seguenti tipi di dati sono supportati da Amazon Kinesis Video Signaling Channels:

- [IceServer](#)

I seguenti tipi di dati sono supportati da Amazon Kinesis Video WebRTC Storage:

Flusso di video Amazon Kinesis

I seguenti tipi di dati sono supportati da Amazon Kinesis Video Streams:

- [ChannelInfo](#)
- [ChannelNameCondition](#)
- [DeletionConfig](#)
- [EdgeAgentStatus](#)
- [EdgeConfig](#)
- [ImageGenerationConfiguration](#)
- [ImageGenerationDestinationConfig](#)
- [LastRecorderStatus](#)
- [LastUploaderStatus](#)
- [ListEdgeAgentConfigurationsEdgeConfig](#)
- [LocalSizeConfig](#)
- [MappedResourceConfigurationListItem](#)
- [MediaSourceConfig](#)
- [MediaStorageConfiguration](#)
- [NotificationConfiguration](#)
- [NotificationDestinationConfig](#)
- [RecorderConfig](#)
- [ResourceEndpointListItem](#)
- [ScheduleConfig](#)
- [SingleMasterChannelEndpointConfiguration](#)
- [SingleMasterConfiguration](#)
- [StreamInfo](#)
- [StreamNameCondition](#)
- [Tag](#)
- [UploaderConfig](#)

ChannelInfo

Servizio: Amazon Kinesis Video Streams

Una struttura che incapsula i metadati e le proprietà di un canale di segnalazione.

Indice

ChannelARN

Il nome della risorsa Amazon (ARN) del canale di segnalazione.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

ChannelName

Il nome del canale di segnalazione.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

ChannelStatus

Stato attuale del canale di segnalazione.

▪Tipo: stringa

Valori validi: `CREATING | ACTIVE | UPDATING | DELETING`

Campo obbligatorio: no

ChannelType

Il tipo di canale di segnalazione.

▪Tipo: stringa

Valori validi: SINGLE_MASTER | FULL_MESH

Campo obbligatorio: no

CreationTime

L'ora in cui è stato creato il canale di segnalazione.

Tipo: Timestamp

Campo obbligatorio: no

SingleMasterConfiguration

Una struttura che contiene la configurazione per il tipo di SINGLE_MASTER canale.

Tipo: oggetto [SingleMasterConfiguration](#)

Campo obbligatorio: no

Version

La versione attuale del canale di segnalazione.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 64 caratteri.

Modello: [a-zA-Z0-9]+

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

ChannelNameCondition

Servizio: Amazon Kinesis Video Streams

Un parametro di input opzionale per l'`ListSignalingChannelsAPI`. Quando questo parametro viene specificato durante l'invocazione `ListSignalingChannels`, l'API restituisce solo i canali che soddisfano una condizione specificata in `ChannelNameCondition`.

Indice

ComparisonOperator

Un operatore di confronto. Attualmente, è possibile specificare solo l'`BEGINS_WITH` operatore, che trova i canali di segnalazione i cui nomi iniziano con un determinato prefisso.

• Tipo: stringa

Valori validi: `BEGINS_WITH`

Campo obbligatorio: no

ComparisonValue

Un valore da confrontare.

• Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

DeletionConfig

Servizio: Amazon Kinesis Video Streams

I dettagli di configurazione necessari per eliminare la connessione dello stream dall'Edge Agent.

Indice

DeleteAfterUpload

Il booleano valore utilizzato per indicare se si desidera contrassegnare o meno il file multimediale per l'eliminazione, una volta caricato sul cloud Kinesis Video Stream. I file multimediali possono essere eliminati se uno qualsiasi dei valori di configurazione di eliminazione è impostato su `true`, ad esempio quando è stato raggiunto il limite di `EdgeRetentionInHoursMaxLocalMediaSizeInMB`, o il.

Poiché il valore predefinito è impostato su `true`, configura la pianificazione dell'uploader in modo che i file multimediali non vengano eliminati prima di essere inizialmente caricati sul AWS cloud.

Tipo: Booleano

Campo obbligatorio: no

EdgeRetentionInHours

Il numero di ore in cui desideri conservare i dati nello stream su Edge Agent. Il valore predefinito del tempo di conservazione è 720 ore, che si traduce in 30 giorni.

Tipo: integer

Intervallo valido: valore minimo di 1. Valore massimo di 720.

Campo obbligatorio: no

LocalSizeConfig

Il valore della dimensione locale richiesta per eliminare la configurazione del bordo.

Tipo: oggetto [LocalSizeConfig](#)

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

EdgeAgentStatus

Servizio: Amazon Kinesis Video Streams

Un oggetto che contiene i dettagli più recenti sullo stato dei lavori di registrazione e caricamento di un agente edge. Utilizza queste informazioni per determinare lo stato attuale di un agente edge.

Indice

LastRecorderStatus

Lo stato più recente del processo di registrazione edge di uno stream.

Tipo: oggetto [LastRecorderStatus](#)

Campo obbligatorio: no

LastUploaderStatus

Lo stato più recente del processo di upload dall'edge al cloud di uno stream.

Tipo: oggetto [LastUploaderStatus](#)

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

EdgeConfig

Servizio: Amazon Kinesis Video Streams

Una descrizione della configurazione edge dello stream che verrà utilizzata per la sincronizzazione con il componente Edge Agent IoT Greengrass. Il componente Edge Agent verrà eseguito su un dispositivo Hub IoT configurato presso la tua sede.

Indice

HubDeviceArn

La «cosa dell'Internet of Things (IoT)» dello stream.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:iot:[a-z0-9-]+:[0-9]+:thing/[a-zA-Z0-9_.-]+`

Campo obbligatorio: sì

RecorderConfig

La configurazione del registratore è costituita dai `MediaSourceConfig` dettagli locali, che vengono utilizzati come credenziali per accedere ai file multimediali locali trasmessi in streaming sulla fotocamera.

Tipo: oggetto [RecorderConfig](#)

Campo obbligatorio: sì

DeletionConfig

La configurazione di eliminazione è costituita dai dettagli relativi al tempo di conservazione (`EdgeRetentionInHours`) e alla configurazione della dimensione locale (`LocalSizeConfig`) utilizzati per effettuare l'eliminazione.

Tipo: oggetto [DeletionConfig](#)

Campo obbligatorio: no

UploaderConfig

La configurazione dell'uploader contiene i `ScheduleExpression` dettagli utilizzati per pianificare i processi di caricamento dei file multimediali registrati da Edge Agent a un Kinesis Video Stream.

Tipo: oggetto [UploaderConfig](#)

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

ImageGenerationConfiguration

Servizio: Amazon Kinesis Video Streams

La struttura che contiene le informazioni necessarie per la distribuzione delle immagini KVS. Se null, la configurazione verrà eliminata dallo stream.

Indice

DestinationConfig

La struttura che contiene le informazioni necessarie per fornire immagini a un cliente.

Tipo: oggetto [ImageGenerationDestinationConfig](#)

Campo obbligatorio: sì

Format

Il formato di immagine accettato.

▪Tipo: stringa

Valori validi: JPEG | PNG

Campo obbligatorio: sì

ImageSelectorType

L'origine dei timestamp del Server o del Produttore da utilizzare per generare le immagini.

▪Tipo: stringa

Valori validi: SERVER_TIMESTAMP | PRODUCER_TIMESTAMP

Campo obbligatorio: sì

SamplingInterval

L'intervallo di tempo in millisecondi (ms) in cui le immagini devono essere generate dallo stream. Il valore minimo che può essere fornito è 200 ms. Se l'intervallo del timestamp è inferiore all'intervallo di campionamento, l'immagine di StartTimestamp verrà restituita, se disponibile.

Tipo: integer

Intervallo valido: valore minimo di 3000. Valore massimo di 20000.

Campo obbligatorio: sì

Status

Indica se l'`ContinuousImageGenerationConfigurationsAPI` è abilitata o disabilitata.

▪Tipo: stringa

Valori validi: `ENABLED` | `DISABLED`

Campo obbligatorio: sì

FormatConfig

L'elenco di una struttura di coppie chiave-valore che contiene parametri aggiuntivi che possono essere applicati quando viene generata l'immagine. La `FormatConfig` chiave è la `JPEGQuality`, che indica la chiave di qualità JPEG da utilizzare per generare l'immagine. Il `FormatConfig` valore accetta int da 1 a 100. Se il valore è 1, l'immagine verrà generata con una qualità inferiore e la compressione migliore. Se il valore è 100, l'immagine verrà generata con la migliore qualità e una compressione inferiore. Se non viene fornito alcun valore, il valore predefinito della `JPEGQuality` chiave verrà impostato su 80.

Tipo: mappatura stringa a stringa

Voci sulla mappa: numero massimo di 1 elemento.

Chiavi valide: `JPEGQuality`

Limiti di lunghezza del valore: lunghezza minima di 0. La lunghezza massima è 256 caratteri.

Modello di valore: `^[a-zA-Z_0-9]+`

Campo obbligatorio: no

HeightPixels

L'altezza dell'immagine di output utilizzata insieme al parametro. `WidthPixels` Quando vengono forniti entrambi `HeightPixels` i `WidthPixels` parametri, l'immagine verrà allungata per adattarsi alle proporzioni specificate. Se viene fornito solo il `HeightPixels` parametro, per calcolare il rapporto verranno utilizzate le `WidthPixels` proporzioni originali. Se non viene fornito alcun parametro, verrà restituita la dimensione originale dell'immagine.

Tipo: integer

Intervallo valido: valore minimo di 1. Valore massimo di 2160.

Campo obbligatorio: no

WidthPixels

La larghezza dell'immagine di output utilizzata insieme al HeightPixels parametro. Quando vengono forniti entrambi WidthPixels e HeightPixels parametri, l'immagine verrà allungata per adattarsi alle proporzioni specificate. Se viene fornito solo il WidthPixels parametro, per calcolare il rapporto verranno utilizzate le HeightPixels proporzioni originali. Se non viene fornito alcun parametro, verrà restituita la dimensione originale dell'immagine.

Tipo: integer

Intervallo valido: valore minimo di 1. Valore massimo di 3840.

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

ImageGenerationDestinationConfig

Servizio: Amazon Kinesis Video Streams

La struttura che contiene le informazioni necessarie per fornire le immagini a un cliente.

Indice

DestinationRegion

La AWS regione del bucket S3 in cui verranno distribuite le immagini. `DestinationRegion` deve corrispondere alla regione in cui si trova lo stream.

▀Tipo: stringa

Vincoli di lunghezza: lunghezza minima di 9. Lunghezza massima di 14.

Modello: `^[a-z]+(-[a-z]+)?-[a-z]+-[0-9]$`

Campo obbligatorio: sì

Uri

L'URI (Uniform Resource Identifier) che identifica dove verranno consegnate le immagini.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 1. Lunghezza massima di 255.

Modello: `^[a-zA-Z_0-9]+:(//)?([/]+)/?([^*]*)$`

Campo obbligatorio: sì

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

LastRecorderStatus

Servizio: Amazon Kinesis Video Streams

Lo stato più recente del processo di registrazione edge di uno stream.

Indice

JobStatusDetails

Una descrizione dello stato più recente di un lavoro di registrazione.

▪Tipo: stringa

Campo obbligatorio: no

LastCollectedTime

Il timestamp in cui il processo di registrazione è stato eseguito l'ultima volta e i file multimediali sono stati archiviati sul disco locale.

Tipo: Timestamp

Campo obbligatorio: no

LastUpdatedTime

Il timestamp in cui lo stato del registratore è stato aggiornato l'ultima volta.

Tipo: Timestamp

Campo obbligatorio: no

RecorderStatus

Lo stato dell'ultimo lavoro del registratore.

▪Tipo: stringa

Valori validi: SUCCESS | USER_ERROR | SYSTEM_ERROR

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

LastUploaderStatus

Servizio: Amazon Kinesis Video Streams

Lo stato più recente del processo di upload dall'edge al cloud di uno stream.

Indice

JobStatusDetails

Una descrizione dello stato più recente di un processo di caricamento.

▪Tipo: stringa

Campo obbligatorio: no

LastCollectedTime

Il timestamp in cui il processo di caricamento è stato eseguito l'ultima volta e i file multimediali raccolti nel cloud.

Tipo: Timestamp

Campo obbligatorio: no

LastUpdatedTime

Il timestamp in cui lo stato dell'uploader è stato aggiornato l'ultima volta.

Tipo: Timestamp

Campo obbligatorio: no

UploaderStatus

Lo stato dell'ultimo lavoro da caricatore.

▪Tipo: stringa

Valori validi: SUCCESS | USER_ERROR | SYSTEM_ERROR

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

ListEdgeAgentConfigurationsEdgeConfig

Servizio: Amazon Kinesis Video Streams

Una descrizione della configurazione edge di un singolo flusso.

Indice

CreationTime

Il timestamp in cui lo stream ha creato per la prima volta la configurazione edge.

Tipo: Timestamp

Campo obbligatorio: no

EdgeConfig

Una descrizione della configurazione edge dello stream che verrà utilizzata per la sincronizzazione con il componente Edge Agent IoT Greengrass. Il componente Edge Agent verrà eseguito su un dispositivo Hub IoT configurato presso la tua sede.

Tipo: oggetto [EdgeConfig](#)

Campo obbligatorio: no

FailedStatusDetails

Una descrizione dello stato di errore generato.

▪Tipo: stringa

Campo obbligatorio: no

LastUpdatedTime

Il timestamp dell'ultima volta che lo stream ha aggiornato la configurazione edge.

Tipo: Timestamp

Campo obbligatorio: no

StreamARN

Il nome della risorsa Amazon (ARN) del flusso.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome del flusso .

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

SyncStatus

Lo stato di sincronizzazione corrente della configurazione edge dello stream.

▀Tipo: stringa

Valori validi: SYNCING | ACKNOWLEDGED | IN_SYNC | SYNC_FAILED | DELETING | DELETE_FAILED | DELETING_ACKNOWLEDGED

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

LocalSizeConfig

Servizio: Amazon Kinesis Video Streams

I dettagli di configurazione che includono la dimensione massima del file multimediale (`MaxLocalMediaSizeInMB`) che si desidera archiviare per uno stream su Edge Agent, nonché la strategia da utilizzare (`StrategyOnFullSize`) quando viene raggiunta la dimensione massima di uno stream.

Indice

MaxLocalMediaSizeInMB

La dimensione massima complessiva del file multimediale che si desidera archiviare per uno streaming su Edge Agent.

Tipo: integer

Intervallo valido: valore minimo di 64. Valore massimo di 2000000.

Campo obbligatorio: no

StrategyOnFullSize

La strategia da eseguire quando viene raggiunto il `MaxLocalMediaSizeInMB` limite di uno stream.

▪Tipo: stringa

Valori validi: `DELETE_OLDEST_MEDIA` | `DENY_NEW_MEDIA`

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

MappedResourceConfigurationListItem

Servizio: Amazon Kinesis Video Streams

Una struttura che incapsula o contiene le proprietà di configurazione dell'archiviazione multimediale.

Indice

ARN

L'Amazon Resource Name (ARN) della risorsa Kinesis Video Stream, associata allo stream.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

Type

Il tipo di risorsa associata per lo stream video kinesis.

▪Tipo: stringa

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

MediaSourceConfig

Servizio: Amazon Kinesis Video Streams

I dettagli di configurazione che consistono nelle credenziali richieste (`MediaUriSecretArn` e `MediaUriType`) per accedere ai file multimediali trasmessi in streaming alla telecamera.

Indice

MediaUriSecretArn

L'ARN di AWS Secrets Manager per il nome utente e la password della videocamera o la posizione di un file multimediale locale.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 20. La lunghezza massima è 2048 caratteri.

Modello: `arn:[a-z\d-]+:secretsmanager:[a-z0-9-]+:[0-9]+:secret:[a-zA-Z0-9_.-]+`

Campo obbligatorio: sì

MediaUriType

Il tipo URI (Uniform Resource Identifier). Il `FILE_URI` valore può essere utilizzato per lo streaming di file multimediali locali.

Note

Preview supporta solo il formato URI della sorgente `RTSP_URI` multimediale.

▪Tipo: stringa

Valori validi: `RTSP_URI` | `FILE_URI`

Campo obbligatorio: sì

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

MediaStorageConfiguration

Servizio: Amazon Kinesis Video Streams

Una struttura che incapsula o contiene le proprietà di configurazione dell'archiviazione multimediale.

- Se `StorageStatus` è abilitata, i dati verranno archiviati nell'apposito spazio. `StreamARN` Affinché `WebRTC Ingestion` funzioni, lo stream deve avere la conservazione dei dati abilitata.
- Se `StorageStatus` è disabilitato, non verrà memorizzato alcun dato e il `StreamARN` parametro non sarà necessario.

Indice

Status

Lo stato della configurazione dell'archiviazione multimediale.

▀Tipo: stringa

Valori validi: ENABLED | DISABLED

Campo obbligatorio: sì

StreamARN

Il nome della risorsa Amazon (ARN) del flusso.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)

- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

NotificationConfiguration

Servizio: Amazon Kinesis Video Streams

Usa questa API per configurare le notifiche di Amazon Simple Notification Service (Amazon SNS) per quando i frammenti diventano disponibili in uno stream. Se questo parametro è nullo, la configurazione verrà eliminata dallo stream.

Per ulteriori informazioni, consulta [Notifiche in Kinesis Video Streams](#).

Indice

DestinationConfig

Le informazioni sulla destinazione necessarie per inviare una notifica a un cliente.

Tipo: oggetto [NotificationDestinationConfig](#)

Campo obbligatorio: sì

Status

Indica se una configurazione di notifica è abilitata o disabilitata.

▪Tipo: stringa

Valori validi: ENABLED | DISABLED

Campo obbligatorio: sì

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

NotificationDestinationConfig

Servizio: Amazon Kinesis Video Streams

La struttura che contiene le informazioni necessarie per inviare una notifica a un cliente.

Indice

Uri

L'URI (Uniform Resource Identifier) che identifica dove verranno consegnate le immagini.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 1. Lunghezza massima di 255.

Modello: `^[a-zA-Z_0-9]+:(//)?([/]+)/?([^*]*)$`

Campo obbligatorio: sì

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

RecorderConfig

Servizio: Amazon Kinesis Video Streams

La configurazione del registratore è costituita dai `MediaSourceConfig` dettagli locali che vengono utilizzati come credenziali per accedere ai file multimediali locali trasmessi in streaming sulla fotocamera.

Indice

MediaSourceConfig

I dettagli di configurazione che consistono nelle credenziali richieste (`MediaUriSecretArneMediaUriType`) per accedere ai file multimediali trasmessi in streaming alla telecamera.

Tipo: oggetto [MediaSourceConfig](#)

Campo obbligatorio: sì

ScheduleConfig

La configurazione composta da `ScheduleExpression` e `DurationInMinutes` dettagli che specificano la pianificazione per la registrazione da una videocamera o da un file multimediale locale su Edge Agent. Se l'`ScheduleExpression` attributo non viene fornito, l'Edge Agent sarà sempre impostato sulla modalità di registrazione.

Tipo: oggetto [ScheduleConfig](#)

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

ResourceEndpointListItem

Servizio: Amazon Kinesis Video Streams

Un oggetto che descrive l'endpoint del canale di segnalazione restituito dall'`GetSignalingChannelEndpointAPI`.

L'endpoint del server multimediale corrisponderà al protocollo. `WEBRTC`

Indice

Protocol

Il protocollo del canale di segnalazione restituito dall'`GetSignalingChannelEndpointAPI`.

▪Tipo: stringa

Valori validi: `WSS` | `HTTPS` | `WEBRTC`

Campo obbligatorio: no

ResourceEndpoint

L'endpoint del canale di segnalazione restituito dall'API. `GetSignalingChannelEndpoint`

▪Tipo: stringa

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

ScheduleConfig

Servizio: Amazon Kinesis Video Streams

Questa API consente di specificare la durata della registrazione della fotocamera o del file multimediale locale sull'Edge Agent. `ScheduleConfig` è composto da `ScheduleExpression` e dagli `DurationInMinutes` attributi.

Se non `ScheduleConfig` è fornito in `RecorderConfig`, l'Edge Agent sarà sempre impostato sulla modalità di registrazione.

Se non `ScheduleConfig` è fornito in `UploaderConfig`, l'Edge Agent verrà caricato a intervalli regolari (ogni 1 ora).

Indice

DurationInSeconds

La durata totale della registrazione del file multimediale. Se viene fornito l'`ScheduleExpression` attributo, è necessario specificare anche l'`DurationInSeconds` attributo.

Tipo: integer

Intervallo valido: valore minimo pari a 60. Valore massimo di 3600.

Campo obbligatorio: sì

ScheduleExpression

L'espressione cron Quartz che si occupa della pianificazione dei lavori da registrare dalla fotocamera o da un file multimediale locale sull'Edge Agent. Se non `ScheduleExpression` viene fornito in `RecorderConfig`, l'Edge Agent sarà sempre impostato sulla modalità di registrazione.

Per ulteriori informazioni su Quartz, consulta la pagina [Cron Trigger Tutorial](#) per comprendere le espressioni valide e il loro utilizzo.

Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 11. Lunghezza massima di 100.

Modello: `[^\n]{11,100}`

Campo obbligatorio: sì

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

SingleMasterChannelEndpointConfiguration

Servizio: Amazon Kinesis Video Streams

Un oggetto che contiene la configurazione dell'endpoint per il tipo di SINGLE_MASTER canale.

Indice

Protocols

Questa proprietà viene utilizzata per determinare la natura della comunicazione su questo canale di SINGLE_MASTER segnalazione. Se WSS specificato, questa API restituisce un endpoint websocket. Se HTTPS specificato, questa API restituisce un HTTPS endpoint.

Tipo: matrice di stringhe

Membri dell'array: numero minimo di 1 elemento. Numero massimo 5 elementi.

Valori validi: WSS | HTTPS | WEBRTC

Campo obbligatorio: no

Role

Questa proprietà viene utilizzata per determinare le autorizzazioni di messaggistica in questo canale di SINGLE_MASTER segnalazione. Se MASTER specificato, questa API restituisce un endpoint che un cliente può utilizzare per ricevere offerte e inviare risposte a tutti gli spettatori su questo canale di segnalazione. Se VIEWER specificato, questa API restituisce un endpoint che un client può utilizzare solo per inviare offerte a un altro MASTER client su questo canale di segnalazione.

▪Tipo: stringa

Valori validi: MASTER | VIEWER

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)

- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

SingleMasterConfiguration

Servizio: Amazon Kinesis Video Streams

Una struttura che contiene la configurazione per il tipo di SINGLE_MASTER canale.

Indice

MessageTtlSeconds

Periodo di tempo (in secondi) in cui un canale di segnalazione conserva i messaggi non recapitati prima che vengano eliminati. Utilizzare per aggiornare [UpdateSignalingChannel](#) questo valore.

Tipo: integer

Intervallo valido: valore minimo di 5. Valore massimo di 120.

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

StreamInfo

Servizio: Amazon Kinesis Video Streams

Un oggetto che descrive un flusso video Kinesis.

Indice

CreationTime

Un timestamp che indica quando è stato creato lo stream.

Tipo: Timestamp

Campo obbligatorio: no

DataRetentionInHours

Il periodo di tempo per cui il flusso conserva i dati, in ore.

Tipo: integer

Intervallo valido: valore minimo di 0.

Campo obbligatorio: no

DeviceName

Il nome del dispositivo associato al flusso.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

KmsKeyId

L'ID della chiave AWS Key Management Service (AWS KMS) utilizzata da Kinesis Video Streams per crittografare i dati sullo stream.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 1. La lunghezza massima è 2048 caratteri.

Modello: .+

Campo obbligatorio: no

MediaType

Il MediaType del flusso.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: `[\w\-\.\+]+/[\w\-\.\+]+(,[\w\-\.\+]+/[\w\-\.\+]+)*`

Campo obbligatorio: no

Status

Lo stato dello stream.

▪Tipo: stringa

Valori validi: `CREATING | ACTIVE | UPDATING | DELETING`

Campo obbligatorio: no

StreamARN

Il nome della risorsa Amazon (ARN) del flusso.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 1024 caratteri.

Modello: `arn:[a-z\d-]+:kinesisvideo:[a-z0-9-]+:[0-9]+:[a-z]+/[a-zA-Z0-9_.-]+/[0-9]+`

Campo obbligatorio: no

StreamName

Il nome del flusso .

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Version

Versione del flusso.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 64 caratteri.

Modello: [a-zA-Z0-9]+

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

StreamNameCondition

Servizio: Amazon Kinesis Video Streams

Specifica la condizione che gli stream devono soddisfare per essere restituiti quando elenchi gli stream (vedi l'`ListStreamsAPI`). Una condizione ha un'operazione di confronto e un valore. Attualmente, è possibile specificare solo l'`BEGINS_WITH`operatore, che trova i flussi i cui nomi iniziano con un determinato prefisso.

Indice

ComparisonOperator

Un operatore di confronto. Attualmente è possibile specificare solo l'`BEGINS_WITH`operatore, che trova i flussi i cui nomi iniziano con un determinato prefisso.

▪Tipo: stringa

Valori validi: `BEGINS_WITH`

Campo obbligatorio: no

ComparisonValue

Un valore da confrontare.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: `[a-zA-Z0-9_.-]+`

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

Tag

Servizio: Amazon Kinesis Video Streams

Una coppia chiave-valore associata al canale di segnalazione specificato.

Indice

Key

La chiave del tag associata al canale di segnalazione specificato.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: $^([\backslash\{L\}\backslash\{Z\}\backslash\{N\}_\cdot :/=+\backslash-@]^*)\$$

Campo obbligatorio: sì

Value

Il valore del tag associato al canale di segnalazione specificato.

▀Tipo: stringa

Limitazioni di lunghezza: lunghezza minima di 0. La lunghezza massima è 256 caratteri.

Modello: $[\backslash\{L\}\backslash\{Z\}\backslash\{N\}_\cdot :/=+\backslash-@]^*$

Campo obbligatorio: sì

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

UploaderConfig

Servizio: Amazon Kinesis Video Streams

La configurazione composta da `ScheduleExpression` e dai `DurationInMinutes` dettagli che specificano la pianificazione per la registrazione da una videocamera o da un file multimediale locale sull'Edge Agent. Se non `ScheduleConfig` è fornito in `UploaderConfig`, l'Edge Agent verrà caricato a intervalli regolari (ogni 1 ora).

Indice

ScheduleConfig

La configurazione composta da `ScheduleExpression` e i `DurationInMinutes` dettagli che specificano la pianificazione per la registrazione da una videocamera o da un file multimediale locale su Edge Agent. Se non `ScheduleConfig` viene fornito in questo documento `UploaderConfig`, l'Edge Agent verrà caricato a intervalli regolari (ogni 1 ora).

Tipo: oggetto [ScheduleConfig](#)

Campo obbligatorio: sì

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

i supporti per Amazon Kinesis Video Streams

I seguenti tipi di dati sono supportati da Amazon Kinesis Video Streams Media:

- [StartSelector](#)

StartSelector

Servizio: Amazon Kinesis Video Streams Media

Identifica il blocco del flusso video Kinesis in cui desideri che l'GetMediaAPI inizi a restituire i dati multimediali. Sono disponibili le seguenti opzioni per identificare il blocco iniziale:

- Scegli il blocco più recente (o più vecchio).
- Identifica un blocco specifico. È possibile identificare un blocco specifico fornendo un numero di frammento o un timestamp (server o produttore).
- I metadati di ogni blocco includono un token di continuazione come tag Matroska (MKV) (). `AWS_KINESISVIDEO_CONTINUATION_TOKEN` Se la tua GetMedia richiesta precedente è terminata, puoi utilizzare questo valore di tag nella richiesta successiva. GetMedia L'API inizia quindi a restituire blocchi a partire dal punto in cui è terminata l'ultima API.

Indice

StartSelectorType

Identifica il frammento del flusso video Kinesis da cui desideri iniziare a ottenere i dati.

- `ORA` - Inizia con l'ultima parte dello stream.
- `EARLEST` - Inizia con la prima parte disponibile dello stream.
- `FRAGMENT_NUMBER` - Inizia con il blocco dopo un frammento specifico. È inoltre necessario specificare il parametro `AfterFragmentNumber`.
- `PRODUCER_TIMESTAMP` o `SERVER_TIMESTAMP`: inizia con il blocco contenente un frammento con il timestamp del produttore o del server specificato. Si specifica `StartTimestamp` il timestamp aggiungendo.
- `CONTINUATION_TOKEN` - Leggi utilizzando il token di continuazione specificato.

Note

Se scegli `NOW`, `EARLIEST` o `CONTINUATION_TOKEN` come `startSelectorType`, non fornisci alcuna informazione aggiuntiva in `startSelector`

▪Tipo: stringa

Valori validi: FRAGMENT_NUMBER | SERVER_TIMESTAMP | PRODUCER_TIMESTAMP | NOW | EARLIEST | CONTINUATION_TOKEN

Campo obbligatorio: sì

AfterFragmentNumber

Specificate il numero del frammento da cui desiderate che l'GetMediaAPI inizi a restituire i frammenti.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: `^[0-9]+$`

Campo obbligatorio: no

ContinuationToken

Token di continuazione restituito da Kinesis Video Streams nella risposta precedente. GetMedia L'GetMediaAPI inizia quindi con il blocco identificato dal token di continuazione.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: `^[a-zA-Z0-9_\. \-]+$`

Campo obbligatorio: no

StartTimestamp

Un valore di timestamp. Questo valore è obbligatorio se si sceglie PRODUCER_TIMESTAMP o SERVER_TIMESTAMP come. startSelectorType L'GetMediaAPI inizia quindi con il blocco contenente il frammento con il timestamp specificato.

Tipo: Timestamp

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli SDK specifici della lingua, consulta AWS quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

Amazon Kinesis Video Streams

I seguenti tipi di dati sono supportati da Amazon Kinesis Video Streams Archived Media:

- [ClipFragmentSelector](#)
- [ClipTimestampRange](#)
- [DASHFragmentSelector](#)
- [DASHTimestampRange](#)
- [Fragment](#)
- [FragmentSelector](#)
- [HLSFragmentSelector](#)
- [HLSTimestampRange](#)
- [Image](#)
- [TimestampRange](#)

ClipFragmentSelector

Servizio: Amazon Kinesis Video Streams Archived Media

Descrive l'intervallo di timestamp e l'origine del timestamp di un intervallo di frammenti.

I frammenti con timestamp del produttore duplicati vengono deduplicati. Ciò significa che se i produttori producono un flusso di frammenti con timestamp del produttore approssimativamente uguali all'ora effettiva, la clip conterrà tutti i frammenti compresi nell'intervallo di timestamp richiesto. Se alcuni frammenti vengono importati nello stesso intervallo di tempo e in momenti molto diversi, viene restituita solo la raccolta di frammenti più vecchia.

Indice

FragmentSelectorType

L'origine dei timestamp da utilizzare (Server o Producer).

▀Tipo: stringa

Valori validi: PRODUCER_TIMESTAMP | SERVER_TIMESTAMP

Campo obbligatorio: sì

TimestampRange

L'intervallo di timestamp da restituire.

Tipo: oggetto [ClipTimestampRange](#)

Campo obbligatorio: sì

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

ClipTimestampRange

Servizio: Amazon Kinesis Video Streams Archived Media

L'intervallo di timestamp per cui restituire i frammenti.

Indice

EndTimeStamp

La fine dell'intervallo di timestamp per il supporto richiesto.

Questo valore deve essere compreso tra 24 ore e deve essere successivo al `StartTimeStamp` valore specificato `StartTimeStamp`. Se `FragmentSelectorType` per la richiesta è `SERVER_TIMESTAMP`, questo valore deve appartenere al passato.

Questo valore è inclusivo. `EndTimeStamp`Viene confrontato con il timestamp (iniziale) del frammento. I frammenti che iniziano prima del `EndTimeStamp` valore e continuano oltre il valore vengono inclusi nella sessione.

Tipo: Timestamp

Campo obbligatorio: sì

StartTimeStamp

Il timestamp di inizio nell'intervallo di timestamp per cui restituire i frammenti.

Nella sessione vengono inclusi solo i frammenti che iniziano esattamente con o dopo `StartTimeStamp`. I frammenti che iniziano prima `StartTimeStamp` e continuano dopo non sono inclusi nella sessione. In caso `FragmentSelectorType` `SERVER_TIMESTAMP` affermativo, `StartTimeStamp` deve essere successivo all'inizio dello stream.

Tipo: Timestamp

Campo obbligatorio: sì

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)

- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

DASHFragmentSelector

Servizio: Amazon Kinesis Video Streams Archived Media

Contiene l'intervallo di timestamp per i file multimediali richiesti e la fonte dei timestamp.

Indice

FragmentSelectorType

L'origine dei timestamp per il supporto richiesto.

[Quando FragmentSelectorType è impostato su PRODUCER_TIMESTAMP e getDash StreamingSession URL: PlaybackMode is ON_DEMAND or LIVE_REPLAY, il primo frammento importato con un timestamp del produttore entro il valore specificato: viene incluso nella playlist multimediale. FragmentSelector TimestampRange Inoltre, sono inclusi i frammenti con data e ora del produttore all'interno di quello TimestampRange importato immediatamente dopo il primo frammento \(fino all'URL getDash: value\). StreamingSession MaxManifestFragmentResults](#)

I frammenti con timestamp del produttore duplicati vengono deduplicati. Ciò significa che se i produttori producono un flusso di frammenti con timestamp del produttore approssimativamente uguali all'ora effettiva, il manifesto MPEG-DASH conterrà tutti i frammenti compresi nell'intervallo di timestamp richiesto. Se alcuni frammenti vengono importati nello stesso intervallo di tempo e in momenti molto diversi, viene restituita solo la raccolta di frammenti più vecchia.

Quando FragmentSelectorType è impostato su PRODUCER_TIMESTAMP e [getDash StreamingSession URL: PlaybackMode isLIVE](#), i timestamp del produttore vengono utilizzati nei frammenti MP4 e per la deduplicazione. Ma i frammenti importati più di recente in base ai timestamp del server sono inclusi nel manifesto MPEG-DASH. Ciò significa che anche se i frammenti ingeriti in passato hanno timestamp del produttore con valori attuali, non sono inclusi nella playlist multimediale HLS.

Il valore predefinito è SERVER_TIMESTAMP.

▀Tipo: stringa

Valori validi: PRODUCER_TIMESTAMP | SERVER_TIMESTAMP

Campo obbligatorio: no

TimestampRange

L'inizio e la fine dell'intervallo di timestamp per il file multimediale richiesto.

Questo valore non dovrebbe essere presente se PlaybackType è LIVE

Tipo: oggetto [DASHTimestampRange](#)

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

DASHTimestampRange

Servizio: Amazon Kinesis Video Streams Archived Media

L'inizio e la fine dell'intervallo di timestamp per il supporto richiesto.

Questo valore non dovrebbe essere presente se PlaybackType è LIVE.

I valori inseriti DASHTimestampRange sono inclusivi. I frammenti che iniziano esattamente all'ora di inizio o dopo l'ora di inizio sono inclusi nella sessione. I frammenti che iniziano prima dell'ora di inizio e continuano oltre tale ora non sono inclusi nella sessione.

Indice

EndTimeStamp

La fine dell'intervallo di timestamp per il supporto richiesto. Questo valore deve essere compreso tra 24 ore e deve essere successivo al StartTimestamp valore specificato StartTimestamp.

Se FragmentSelectorType per la richiesta è SERVER_TIMESTAMP, questo valore deve appartenere al passato.

Il EndTimestamp valore è obbligatorio per la ON_DEMAND modalità, ma facoltativo per la LIVE_REPLAY modalità. Se la LIVE_REPLAY modalità non EndTimestamp è impostata, la sessione continuerà a includere i nuovi frammenti ingeriti fino alla scadenza della sessione.

Note

Questo valore è inclusivo. EndTimestampViene confrontato con il timestamp (iniziale) del frammento. I frammenti che iniziano prima del EndTimestamp valore e continuano oltre il valore vengono inclusi nella sessione.

Tipo: Timestamp

Campo obbligatorio: no

StartTimestamp

L'inizio dell'intervallo di timestamp per il supporto richiesto.

Se il DASHTimestampRange valore è specificato, il StartTimestamp valore è obbligatorio.

Nella sessione `StartTimeStamp` vengono inclusi solo i frammenti che iniziano esattamente con o dopo. I frammenti che iniziano prima `StartTimeStamp` e continuano dopo non sono inclusi nella sessione. In caso `FragmentSelectorType SERVER_TIMESTAMP` affermativo, `StartTimeStamp` deve essere successivo all'inizio dello stream.

Tipo: `Timestamp`

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

Fragment

Servizio: Amazon Kinesis Video Streams Archived Media

Rappresenta un segmento di video o altri dati delimitati nel tempo.

Indice

FragmentLengthInMilliseconds

La durata della riproduzione o altro valore temporale associato al frammento.

Tipo: long

Campo obbligatorio: no

FragmentNumber

L'identificatore univoco del frammento. Questo valore aumenta in modo monotono in base all'ordine di ingestione.

■Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 128 caratteri.

Modello: $^{\wedge}[0-9]^{\wedge}+$

Campo obbligatorio: no

FragmentSizeInBytes

La dimensione totale del frammento, comprese le informazioni sul frammento e i dati multimediali contenuti.

Tipo: long

Campo obbligatorio: no

ProducerTimestamp

Il timestamp del produttore corrispondente al frammento, in millisecondi.

Tipo: Timestamp

Campo obbligatorio: no

ServerTimestamp

Il timestamp del AWS server corrispondente al frammento, in millisecondi.

Tipo: Timestamp

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli SDK specifici della lingua AWS , consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

FragmentSelector

Servizio: Amazon Kinesis Video Streams Archived Media

Descrive l'intervallo di timestamp e l'origine del timestamp di un intervallo di frammenti.

Vengono restituiti solo i frammenti con un timestamp di inizio maggiore o uguale all'ora di inizio specificata e minore o uguale all'ora di fine. Ad esempio, se uno stream contiene frammenti con i seguenti timestamp di inizio:

- 00:00:00
- 00:00:02
- 00:00:04
- 00:00:06

Un intervallo di selettori di frammenti con un'ora di inizio di 00:00:01 e un'ora di fine di 00:00:04 restituirebbe i frammenti con orari di inizio 00:00:02 e 00:00:04.

Indice

FragmentSelectorType

L'origine dei timestamp da utilizzare (Server o Producer).

▪Tipo: stringa

Valori validi: PRODUCER_TIMESTAMP | SERVER_TIMESTAMP

Campo obbligatorio: sì

TimestampRange

L'intervallo di timestamp da restituire.

Tipo: oggetto [TimestampRange](#)

Campo obbligatorio: sì

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

HLSFragmentSelector

Servizio: Amazon Kinesis Video Streams Archived Media

Contiene l'intervallo di timestamp per i file multimediali richiesti e la fonte dei timestamp.

Indice

FragmentSelectorType

L'origine dei timestamp per il supporto richiesto.

[Quando FragmentSelectorType è impostato su PRODUCER_TIMESTAMP e getHLS StreamingSession URL: PlaybackMode is ON_DEMAND or LIVE_REPLAY, il primo frammento importato con un timestamp del produttore entro il valore specificato FragmentSelector: viene incluso nella playlist multimediale. TimestampRange Inoltre, sono inclusi i frammenti con data e ora del produttore all'interno di quello TimestampRange importato immediatamente dopo il primo frammento \(fino all'URL di GetHLS: value\). StreamingSession MaxMediaPlaylistFragmentResults](#)

I frammenti con timestamp del produttore duplicati vengono deduplicati. Ciò significa che se i produttori producono un flusso di frammenti con timestamp del produttore approssimativamente uguali all'ora effettiva, le playlist multimediali HLS conterranno tutti i frammenti compresi nell'intervallo di timestamp richiesto. Se alcuni frammenti vengono importati nello stesso intervallo di tempo e in momenti molto diversi, viene restituita solo la raccolta di frammenti più vecchia.

Quando FragmentSelectorType è impostato su PRODUCER_TIMESTAMP e [getHLS StreamingSession URL: PlaybackMode is LIVE](#), i timestamp del produttore vengono utilizzati nei frammenti MP4 e per la deduplicazione. Ma i frammenti importati più di recente in base ai timestamp del server sono inclusi nella playlist multimediale HLS. Ciò significa che anche se i frammenti ingeriti in passato hanno timestamp del produttore con valori attuali, non sono inclusi nella playlist multimediale HLS.

Il valore predefinito è SERVER_TIMESTAMP.

•Tipo: stringa

Valori validi: PRODUCER_TIMESTAMP | SERVER_TIMESTAMP

Campo obbligatorio: no

TimestampRange

L'inizio e la fine dell'intervallo di timestamp per il file multimediale richiesto.

Questo valore non dovrebbe essere presente se PlaybackType è LIVE

Tipo: oggetto [HLSTimestampRange](#)

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

HLSTimestampRange

Servizio: Amazon Kinesis Video Streams Archived Media

L'inizio e la fine dell'intervallo di timestamp per il supporto richiesto.

Questo valore non dovrebbe essere presente se PlaybackType è LIVE.

Indice

EndTimeStamp

La fine dell'intervallo di timestamp per il supporto richiesto. Questo valore deve essere compreso tra 24 ore e deve essere successivo al StartTimestamp valore specificato StartTimestamp.

Se FragmentSelectorType per la richiesta è SERVER_TIMESTAMP, questo valore deve appartenere al passato.

Il EndTimestamp valore è obbligatorio per la ON_DEMAND modalità, ma facoltativo per la LIVE_REPLAY modalità. Se la LIVE_REPLAY modalità non EndTimestamp è impostata, la sessione continuerà a includere i nuovi frammenti ingeriti fino alla scadenza della sessione.

Note

Questo valore è inclusivo. EndTimestampViene confrontato con il timestamp (iniziale) del frammento. I frammenti che iniziano prima del EndTimestamp valore e continuano oltre il valore vengono inclusi nella sessione.

Tipo: Timestamp

Campo obbligatorio: no

StartTimestamp

L'inizio dell'intervallo di timestamp per il supporto richiesto.

Se il HLSTimestampRange valore è specificato, il StartTimestamp valore è obbligatorio.

Nella sessione StartTimestamp vengono inclusi solo i frammenti che iniziano esattamente con o dopo. I frammenti che iniziano prima StartTimestamp e continuano dopo non sono inclusi nella sessione. In caso FragmentSelectorType SERVER_TIMESTAMP affermativo, StartTimestamp deve essere successivo all'inizio dello stream.

Tipo: Timestamp

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

Image

Servizio: Amazon Kinesis Video Streams Archived Media

Una struttura che contiene `TimestampError`, `eImageContent`.

Indice

Error

Il messaggio di errore visualizzato quando l'immagine per il timestamp fornito non è stata estratta a causa di un errore non verificabile. Verrà restituito un errore se:

- Non esiste alcun supporto per il file specificato `Timestamp`.
- Il supporto per il periodo di tempo specificato non consente l'estrazione di un'immagine. In questo caso il supporto è solo audio oppure è stato inserito il supporto errato.

■Tipo: stringa

Valori validi: `NO_MEDIA` | `MEDIA_ERROR`

Campo obbligatorio: no

ImageContent

Un attributo dell'Imageoggetto codificato in Base64.

■Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. Lunghezza massima di 6291456.

Campo obbligatorio: no

TimeStamp

Un attributo dell'Imageoggetto utilizzato per estrarre un'immagine dal flusso video. Questo campo viene utilizzato per gestire le lacune nelle immagini o per comprendere meglio la finestra di impaginazione.

Tipo: Timestamp

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

TimestampRange

Servizio: Amazon Kinesis Video Streams Archived Media

L'intervallo di timestamp per cui restituire i frammenti.

Indice

EndTimestamp

Il timestamp di fine nell'intervallo di timestamp per cui restituire i frammenti.

Tipo: Timestamp

Campo obbligatorio: sì

StartTimestamp

Il timestamp di inizio nell'intervallo di timestamp per cui restituire i frammenti.

Tipo: Timestamp

Campo obbligatorio: sì

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli SDK specifici della lingua AWS , consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

Flesso di video Amazon Kinesis

I seguenti tipi di dati sono supportati dai canali di segnalazione video di Amazon Kinesis:

- [IceServer](#)

IceServer

Servizio: Amazon Kinesis Video Signaling Channels

Una struttura per i dati di connessione al server ICE.

Indice

Password

Una password per accedere al server ICE.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_ . -]+

Campo obbligatorio: no

Ttl

Periodo di tempo, in secondi, durante il quale il nome utente e la password sono validi.

Tipo: integer

Intervallo valido: valore minimo di 30. Valore massimo pari a 86400.

Campo obbligatorio: no

Uris

Una matrice di URI, nel formato specificato nell'[I-D. petithuguenin-behave-turn-urisspec](#). Questi URI forniscono i diversi indirizzi e/o protocolli che possono essere utilizzati per raggiungere il server TURN.

Tipo: matrice di stringhe

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Campo obbligatorio: no

Username

Un nome utente per accedere al server ICE.

▪Tipo: stringa

Limitazioni di lunghezza: lunghezza minima pari a 1. La lunghezza massima è 256 caratteri.

Modello: [a-zA-Z0-9_.-]+

Campo obbligatorio: no

Vedi anche

Per ulteriori informazioni sull'utilizzo di questa API in uno degli AWS SDK specifici della lingua, consulta quanto segue:

- [AWS SDK per C++](#)
- [AWS SDK per Java V2](#)
- [AWS SDK per Ruby V3](#)

Amazon Kinesis Video Video Video Video Video Video Video Video WebRTC
Video Video Video Video Video Video Video Video Video

I seguenti tipi di dati sono supportati da Amazon Kinesis Video WebRTC Storage:

Errori comuni

In questa sezione sono riportati gli errori comuni delle azioni API per tutti i servizi AWS. Per gli errori specifici di un'azione API per questo servizio, consulta l'argomento per quell'azione API.

AccessDeniedException

Non disponi dell'autorizzazione di accesso sufficiente per eseguire questa operazione.

Codice di stato HTTP: 400

IncompleteSignature

La firma della richiesta non è conforme agli standard AWS.

Codice di stato HTTP: 400

InternalFailure

L'elaborazione della richiesta non è riuscita a causa di un errore, un'eccezione o un guasto interno sconosciuto.

Codice di stato HTTP: 500

InvalidAction

L'azione o l'operazione richiesta non è valida. Verifica che l'operazione sia digitata correttamente.

Codice di stato HTTP: 400

InvalidClientTokenId

Il certificato X.509 o l'ID chiave di accesso AWS forniti non sono presenti nei nostri record.

Codice di stato HTTP: 403

NotAuthorized

Non disponi delle autorizzazioni per eseguire questa azione.

Codice di stato HTTP: 400

OptInRequired

L'ID chiave di accesso AWS necessita di una sottoscrizione al servizio.

Codice di stato HTTP: 403

RequestExpired

La richiesta ha raggiunto il servizio più di 15 minuti dopo il date stamp della richiesta o più di 15 minuti dopo la data di scadenza della richiesta (ad esempio per URL prefirmati) oppure il date stamp della richiesta è più di 15 minuti nel futuro.

Codice di stato HTTP: 400

ServiceUnavailable

La richiesta non è riuscita a causa di un errore temporaneo del server.

Codice di stato HTTP: 503

ThrottlingException

La richiesta è stata negata a causa del throttling della richiesta.

Codice di stato HTTP: 400

ValidationError

L'input non riesce a soddisfare i vincoli specificati da un servizio AWS.

Codice di stato HTTP: 400

Parametri comuni

L'elenco seguente contiene i parametri utilizzati da tutte le azioni per firmare le richieste di Signature Version 4 con una stringa di query. Qualsiasi parametro specifico di un'operazione è riportato nell'argomento relativo all'operazione. Per ulteriori informazioni sull'utilizzo di Signature Version 4, consulta la pagina [Firma delle richieste API AWS](#) nella Guida per l'utente di IAM.

Action

azione da eseguire.

Tipo: stringa

Campo obbligatorio: sì

Version

Versione dell'API per cui è scritta la richiesta, espressa nel formato AAAA-MM-GG.

Tipo: stringa

Campo obbligatorio: sì

X-Amz-Algorithm

Algoritmo hash utilizzato per creare la firma della richiesta.

Condition: specifica questo parametro quando includi le informazioni di autenticazione in una stringa di query anziché nell'intestazione di autorizzazione HTTP.

Tipo: stringa

Valori validi: AWS4-HMAC-SHA256

Obbligatorio: condizionale

X-Amz-Credential

Il valore dell'ambito delle credenziali, che è una stringa che include la chiave di accesso, la data, la regione di destinazione, il servizio richiesto e una stringa di terminazione ("aws4_request").

Il valore viene espresso nel seguente formato: chiave_accesso/AAAAMMGG/regione/servizio/aws4_request.

Per ulteriori informazioni, consulta la pagina [Creazione di una richiesta API AWS firmata](#) nella Guida per l'utente di IAM.

Condition: specifica questo parametro quando includi le informazioni di autenticazione in una stringa di query anziché nell'intestazione di autorizzazione HTTP.

Tipo: stringa

Obbligatorio: condizionale

X-Amz-Date

La data utilizzata per creare la firma. Il formato deve essere il formato di base ISO 8601 (YYYYMMDD'THHMMSS'Z'). Ad esempio, la seguente combinazione data/ora è un valore X-Amz-Date valido: 20120325T120000Z.

Condition: X-Amz-Date è facoltativo per tutte le richieste; può essere utilizzato per sovrascrivere la data utilizzata per firmare le richieste. Se l'intestazione Date è specificata nel formato base ISO 8601, X-Amz-Date non è richiesto. Quando utilizzi X-Amz-Date, sostituisce sempre il valore dell'intestazione Date. Per ulteriori informazioni, consulta la pagina [Elementi di una firma di richiesta API AWS](#) nella Guida per l'utente di IAM.

Tipo: stringa

Obbligatorio: condizionale

X-Amz-Security-Token

Il token di sicurezza provvisorio ottenuto tramite una chiamata ad AWS Security Token Service (AWS STS). Per un elenco di servizi che supportano le credenziali di sicurezza temporanee da AWS STS, consulta la pagina [Servizi AWS che funzionano con IAM](#) nella Guida per l'utente di IAM.

Condizione: se utilizzi le credenziali di sicurezza temporanee fornite da AWS STS, devi includere il token di sicurezza.

Tipo: stringa

Obbligatorio: condizionale

X-Amz-Signature

Specifica la firma con codifica esadecimale calcolata dalla stringa da firmare e dalla chiave di firma derivata.

Condition: specifica questo parametro quando includi le informazioni di autenticazione in una stringa di query anziché nell'intestazione di autorizzazione HTTP.

Tipo: stringa

Obbligatorio: condizionale

X-Amz-SignedHeaders

Specifica tutte le intestazioni HTTP incluse come parte della richiesta canonica. Per ulteriori informazioni sulla specifica delle intestazioni firmate, consulta la pagina [Creazione di una richiesta API AWS firmata](#) nella Guida per l'utente di IAM.

Condition: specifica questo parametro quando includi le informazioni di autenticazione in una stringa di query anziché nell'intestazione di autorizzazione HTTP.

Tipo: stringa

Obbligatorio: condizionale

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.