



Guida per gli sviluppatori

AWS IoT FleetWise



AWS IoT FleetWise: 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 l'AWS IoT FleetWise?	1
Vantaggi	2
Casi d'uso	2
Sei nuovo nell'AWS IoT FleetWise?	3
Accesso all'AWS IoT FleetWise	3
Prezzi per l'AWS IoT FleetWise	3
Come FleetWise funziona AWS IoT	3
Concetti chiave	4
Caratteristiche dell'AWS IoT FleetWise	8
Servizi correlati	8
Configurazione dell' AWS IoT FleetWise	10
Configura il tuo Account AWS	10
Registrati per un Account AWS	10
Crea un utente con accesso amministrativo	11
Nozioni di base sull'utilizzo nella console	12
Configurazione delle impostazioni	12
Configurazione delle impostazioni (console)	13
Configura le impostazioni (AWS CLI)	14
Nozioni di base	16
Requisiti	16
Utilizzo della demo del software Edge Agent	16
Nozioni di base (console)	17
Prerequisiti	18
Passaggio 1: configurare il software Edge Agent per AWS IoT FleetWise	18
Fase 2: Creare un modello di veicolo	20
Fase 3: Creare un manifesto del decodificatore	22
Passaggio 4: Configurare un manifesto del decodificatore	23
Fase 5: Creare un veicolo	24
Passaggio 6: crea una campagna	25
Fase 7: pulire	27
Passaggi successivi	27
Inserimento di dati nel cloud	28
Modellazione di veicoli	31
Cataloghi di segnali	34

Configura i segnali	36
Crea un catalogo di segnali (AWS CLI)	43
Importa un catalogo di segnali	47
Aggiornare un catalogo di segnali (AWS CLI)	57
Eliminare un catalogo di segnali (AWS CLI)	59
Ottieni informazioni sul catalogo dei segnali (AWS CLI)	59
Modelli di veicoli	60
Crea un modello di veicolo	61
Aggiornare un modello di veicolo (AWS CLI)	68
Eliminare un modello di veicolo	69
Ottieni informazioni sul modello del veicolo (AWS CLI)	70
Manifesti del decodificatore	71
Configura le interfacce di rete e i segnali del decodificatore	73
Crea un manifesto del decodificatore	75
Aggiornare un manifesto del decoder (AWS CLI)	83
Eliminare un manifesto del decodificatore	84
Ottieni informazioni sul manifesto del decoder (AWS CLI)	85
Veicoli	87
Veicoli di approvvigionamento	88
Autentica i veicoli	89
Autorizza i veicoli	91
Argomenti riservati	92
Crea un veicolo	94
Crea un veicolo (console)	94
Crea un veicolo (AWS CLI)	96
Crea più veicoli (AWS CLI)	98
Aggiorna un veicolo (AWS CLI)	99
Aggiorna più veicoli (AWS CLI)	100
Eliminare un veicolo	101
Eliminare un veicolo (console)	102
Elimina un veicolo (AWS CLI)	102
Ottieni informazioni sul veicolo (AWS CLI)	102
Parchi istanze	104
Crea una flotta (AWS CLI)	105
Associa un veicolo a una flotta (AWS CLI)	106
Dissociare un veicolo da una flotta (AWS CLI)	106

Aggiorna una flotta (AWS CLI)	107
Elimina un parco istanze (AWS CLI)	107
Ottieni informazioni sulla flotta (AWS CLI)	107
Campagne	109
Creazione di una campagna	114
Crea una campagna (console)	115
Crea una campagna () AWS CLI	122
Espressioni logiche per le campagne	126
Aggiorna una campagna (AWS CLI)	127
Eliminare una campagna	128
Eliminare una campagna (console)	128
Eliminare una campagna (AWS CLI)	128
Ottieni informazioni sulla campagna () AWS CLI	128
Elaborazione e visualizzazione dei dati del veicolo	130
Elaborazione dei dati del veicolo in Timestream	130
Visualizzazione dei dati del veicolo memorizzati in Timestream	131
Elaborazione dei dati del veicolo in S3	131
Formato oggetto S3	132
Analisi dei dati del veicolo memorizzati in S3	132
AWS CLI e AWSSDK	135
Risoluzione dei problemi	136
Problemi relativi al manifesto del decoder	136
Problemi relativi al FleetWise software Edge Agent per AWS IoT	140
Problema: il software Edge Agent non si avvia.	140
Problema: [ERRORE] [IoTFleetWiseEngine: :connect]: [Impossibile avviare la libreria di persistenza]	142
Problema: il software Edge Agent non raccoglie i PID di diagnostica di bordo (OBD) II e i codici di errore diagnostici (DTC).	142
Problema: il FleetWise software Edge Agent for AWS IoT non raccoglie dati dalla rete o non è in grado di applicare regole di ispezione dei dati.	142
Problema: [ERROR] [AwsIotConnectivityModule: :connect]: [Connessione fallita con errore] o [WARN] [AwsIotChannel: :send]: [Nessuna connessione MQTT attiva.]	143
Sicurezza	144
Protezione dei dati	145
Crittografia a riposo	146
Crittografia in transito	146

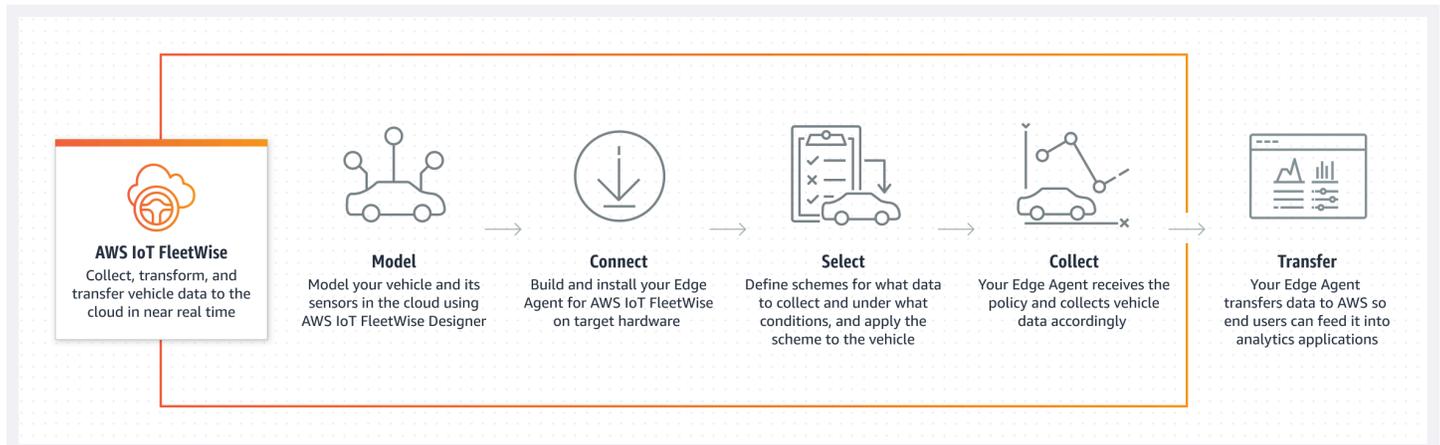
Crittografia dei dati	146
Controllo dell'accesso	155
Concedi AWS IoT FleetWise l'accesso a una destinazione Amazon S3	155
Concedi AWS IoT FleetWise l'accesso a una destinazione Amazon Timestream	158
Identity and Access Management	161
Destinatari	162
Autenticazione con identità	162
Gestione dell'accesso con policy	166
Come FleetWise funziona AWS l'IoT con IAM	169
Esempi di policy basate su identità	178
Risoluzione dei problemi	182
Convalida della conformità	184
Resilienza	185
Sicurezza dell'infrastruttura	186
Connessione all' AWS IoT FleetWise tramite un'interfaccia VPC endpoint	186
Analisi della configurazione e delle vulnerabilità	190
Best practice di sicurezza	190
Concedere autorizzazioni minime possibili	190
Non registrare informazioni riservate	191
Utilizzalo per visualizzare la cronologia AWS CloudTrail delle chiamate API	191
Tenere sincronizzato l'orologio del dispositivo	191
Monitoraggio	192
Monitoraggio con CloudWatch	192
Monitoraggio con log CloudWatch	196
Visualizzazione dei FleetWise log AWS IoT nella console CloudWatch	196
Configurazione della registrazione	201
Log CloudTrail	204
AWSIoT FleetWise informazioni in CloudTrail	205
Comprensione AWSIoT FleetWise voci dei file di registro	206
Cronologia dei documenti	208
.....	CCX

Che cos'è l'AWS IoT FleetWise?

AWS FleetWise L' IoT è un servizio gestito che puoi utilizzare per raccogliere i dati dei veicoli e organizzarli nel cloud. Puoi utilizzare i dati raccolti per migliorare la qualità, le prestazioni e l'autonomia del veicolo. Con l'AWS IoT FleetWise, puoi raccogliere e organizzare dati da veicoli che utilizzano protocolli e formati di dati diversi. AWS L' IoT FleetWise aiuta a trasformare i messaggi di basso livello in valori leggibili dall'uomo e a standardizzare il formato dei dati nel cloud per l'analisi dei dati. Puoi anche definire campagne di raccolta dati per controllare quali dati del veicolo raccogliere e quando trasferirli sul cloud.

Quando i dati dei veicoli sono nel cloud, puoi utilizzarli per applicazioni che analizzano lo stato della flotta di veicoli. Questi dati possono aiutarti a identificare potenziali problemi di manutenzione, rendere più intelligenti i sistemi di infotainment di bordo e migliorare tecnologie avanzate come la guida autonoma e i sistemi di assistenza alla guida con analisi e apprendimento automatico (ML).

Il diagramma seguente mostra l'architettura di base dell'AWS IoT FleetWise.



Argomenti

- [Vantaggi](#)
- [Casi d'uso](#)
- [Sei nuovo nell'AWS IoT FleetWise?](#)
- [Accesso all'AWS IoT FleetWise](#)
- [Prezzi per l'AWS IoT FleetWise](#)
- [Come FleetWise funziona AWS l' IoT](#)
- [Servizi correlati](#)

Vantaggi

I principali vantaggi dell'AWS IoT FleetWise sono:

Raccogli i dati dei veicoli in modo più intelligente

Migliora la pertinenza dei dati con una raccolta intelligente dei dati che invia al cloud solo i dati necessari per l'analisi.

Analizza facilmente dati standardizzati a livello di parco

Analizza i dati standardizzati di una flotta di veicoli senza dover sviluppare un sistema personalizzato di raccolta o registrazione dei dati.

Sincronizzazione automatica dei dati nel cloud

Ottieni una visione unificata dei dati raccolti dai sensori standard (dati di telemetria) e dai sistemi di visione (dati provenienti da telecamere, radar e lidar) e mantienili automaticamente sincronizzati nel cloud. AWS IoT FleetWise mantiene automaticamente sincronizzati nel cloud i dati dei sistemi di visione strutturati e non strutturati, i metadati e i dati dei sensori standard. Ciò semplifica il processo per assemblare una visione d'insieme degli eventi e ottenere informazioni dettagliate.

Note

I dati del sistema di visione sono in versione di anteprima e sono soggetti a modifiche.

Casi d'uso

Gli scenari in cui è possibile utilizzare l'AWS IoT FleetWise includono:

Addestra modelli AI/ML

Migliora continuamente i modelli di apprendimento automatico utilizzati per i sistemi di assistenza alla guida autonomi e avanzati raccogliendo dati dai veicoli di produzione.

Migliora l'esperienza digitale del cliente

Utilizza i dati dei sistemi di infotainment per rendere più pertinenti i contenuti audiovisivi di bordo e gli approfondimenti in-app.

Mantieni la salute della flotta di veicoli

Utilizza le informazioni ricavate dai dati della flotta per monitorare lo stato delle batterie dei veicoli elettrici e i livelli di carica, gestire i programmi di manutenzione, analizzare il consumo di carburante e altro ancora.

Sei nuovo nell'AWSIoT FleetWise?

Se non conosci l'AWSIoT FleetWise, ti consigliamo di iniziare leggendo le seguenti sezioni:

- [Come FleetWise funziona AWS l'IoT](#)
- [Configurazione dell' AWS IoT FleetWise](#)
- [Demo del software Edge Agent](#)
- [Inserimento di dati nel cloud](#)

Accesso all'AWSIoT FleetWise

Puoi utilizzare la FleetWise console o l'API AWS IoT per accedere all'AWSIoT FleetWise.

Prezzi per l'AWSIoT FleetWise

I veicoli inviano dati al cloud tramite messaggi MQTT. Paghi alla fine di ogni mese per i veicoli che hai creato in AWS IoT FleetWise. Paghi anche per i messaggi che raccogli dai veicoli. Per informazioni aggiornate sui prezzi, consulta la pagina [FleetWise dei prezzi AWS IoT](#). Per ulteriori informazioni sul protocollo di messaggistica MQTT, consulta [MQTT](#) nella AWS IoT Core Developer Guide.

Come FleetWise funziona AWS l'IoT

Le seguenti sezioni forniscono una panoramica dei componenti dei FleetWise servizi AWS IoT e del modo in cui interagiscono.

Dopo aver letto questa introduzione, consulta la [Configurazione dell' AWS IoT FleetWise](#) sezione per scoprire come configurare l'AWSIoT FleetWise.

Argomenti

- [Concetti chiave](#)
- [Caratteristiche dell'AWSIoT FleetWise](#)

Concetti chiave

AWS FleetWise L'IoT fornisce un framework di modellazione dei veicoli per modellare il veicolo e i suoi sensori e attuatori nel cloud. Per consentire la comunicazione sicura tra il veicolo e il cloud, l'AWSIoT fornisce FleetWise anche un'implementazione di riferimento per aiutarti a sviluppare il software Edge Agent che puoi installare nel tuo veicolo. Puoi definire schemi di raccolta dati nel cloud e implementarli sul tuo veicolo. Il software Edge Agent in esecuzione sul veicolo utilizza schemi di raccolta dati per controllare quali dati raccogliere e quando trasferirli sul cloud.

Di seguito sono riportati i concetti fondamentali dell'AWSIoT FleetWise.

Signal

I segnali sono strutture fondamentali che definisci per contenere i dati del veicolo e i relativi metadati. Un segnale può essere un attributo, un ramo, un sensore o un attuttore. Ad esempio, è possibile creare un sensore per ricevere i valori di temperatura all'interno del veicolo e memorizzarne i metadati, tra cui il nome del sensore, un tipo di dati e un'unità. Per ulteriori informazioni, consulta [Crea e gestisci cataloghi di segnali](#).

Attributo

Gli attributi rappresentano informazioni statiche che generalmente non cambiano, come il produttore e la data di produzione.

Branch

I rami rappresentano i segnali in una struttura annidata. I rami mostrano le gerarchie dei segnali. Ad esempio, il `Vehicle` ramo ha un ramo figlio, `Powertrain`. Il `Powertrain` ramo ha un ramo figlio, `combustionEngine`. Per localizzare il `combustionEngine` ramo, usa l'espressione `Vehicle.Powertrain.combustionEngine`.

Sensor

I dati del sensore riportano lo stato attuale del veicolo e le variazioni nel tempo, al variare dello stato del veicolo, ad esempio i livelli del fluido, le temperature, le vibrazioni o la tensione.

Actuator

I dati degli attuatori riportano lo stato di un dispositivo del veicolo, come motori, riscaldatori e serrature delle porte. La modifica dello stato di un dispositivo del veicolo può aggiornare i dati dell'attuatore. Ad esempio, è possibile definire un attuatore per rappresentare il riscaldatore. L'attuatore riceve nuovi dati quando si accende o si spegne il riscaldatore.

Struttura personalizzata

Una struttura personalizzata (nota anche come struttura) rappresenta una struttura di dati complessa o di ordine superiore. Facilita l'associazione o il raggruppamento logico di dati provenienti dalla stessa fonte. Una struttura viene utilizzata quando i dati vengono letti o scritti in un'operazione atomica, ad esempio per rappresentare un tipo di dati complesso o una forma di ordine superiore.

Un segnale di tipo di struttura viene definito nel catalogo dei segnali utilizzando un riferimento a un tipo di dati di struttura anziché a un tipo di dati primitivo. Le strutture possono essere utilizzate per tutti i tipi di segnali, inclusi sensori, attributi, attuatori e tipi di dati dei sistemi di visione. Se viene inviato o ricevuto un segnale di tipo struttura, l'AWS IoT FleetWise si aspetta che tutti gli elementi inclusi abbiano valori validi, quindi tutti gli elementi sono obbligatori. Ad esempio, se una struttura contiene gli elementi `vehicle.camera.image.height`, `vehicle.camera.image.width` e `vehicle.camera.image.data`, è previsto che il segnale inviato contenga valori per tutti questi elementi.

Note

I dati del sistema di visione sono in versione di anteprima e sono soggetti a modifiche.

Proprietà personalizzata

Una proprietà personalizzata rappresenta un membro della struttura dati complessa. Il tipo di dati della proprietà può essere primitivo o un'altra struttura.

Quando si rappresenta una forma di ordine superiore utilizzando una struttura e una proprietà personalizzata, la forma di ordine superiore desiderata viene sempre definita e visualizzata come una struttura ad albero. La proprietà personalizzata viene utilizzata per definire tutti i nodi foglia mentre la struttura viene utilizzata per definire tutti i nodi non fogliari.

Catalogo dei segnali

Un catalogo di segnali contiene una raccolta di segnali. I segnali presenti in un catalogo di segnali possono essere utilizzati per modellare veicoli che utilizzano protocolli e formati di dati diversi. Ad esempio, ci sono due auto prodotte da case automobilistiche diverse: una utilizza il protocollo Control Area Network (CAN bus); l'altra utilizza il protocollo OBD (On-board Diagnostics). È possibile definire un sensore nel catalogo dei segnali per ricevere i valori di temperatura all'interno

del veicolo. Questo sensore può essere utilizzato per rappresentare le termocoppie di entrambe le auto. Per ulteriori informazioni, consulta [Crea e gestisci cataloghi di segnali](#).

Modello di veicolo (manifesto del modello)

I modelli di veicolo sono strutture dichiarative che è possibile utilizzare per standardizzare il formato dei veicoli e definire le relazioni tra i segnali nei veicoli. I modelli di veicoli applicano informazioni coerenti su più veicoli dello stesso tipo. Aggiungi segnali per creare modelli di veicoli. Per ulteriori informazioni, consulta [Crea e gestisci modelli di veicoli](#).

Manifesto del decoder

I manifesti del decoder contengono informazioni di decodifica per ogni segnale nei modelli di veicoli. I sensori e gli attuatori dei veicoli trasmettono messaggi di basso livello (dati binari). Con i decoder manifest, l'AWS IoT FleetWise è in grado di trasformare i dati binari in valori leggibili dall'uomo. Ogni manifesto del decoder è associato a un modello di veicolo. Per ulteriori informazioni, consulta [Crea e gestisci i manifesti del decodificatore](#).

Interfaccia di rete

Contiene informazioni sul protocollo utilizzato dalla rete di bordo. AWS IoT FleetWise supporta i seguenti protocolli.

Controller Area Network (bus CAN)

Un protocollo che definisce il modo in cui i dati vengono comunicati tra unità di controllo elettroniche (ECU). Le ECU possono essere l'unità di controllo del motore, gli airbag o il sistema audio.

Diagnostica di bordo (OBD) II

Un protocollo ulteriormente sviluppato che definisce il modo in cui i dati di autodiagnostica vengono comunicati tra le ECU. Fornisce una serie di codici diagnostici di errore (DTC) standard che aiutano a identificare i problemi del veicolo.

Middleware per veicoli

Il middleware del veicolo definito come un tipo di interfaccia di rete. Esempi di middleware per veicoli includono Robot Operating System (ROS 2) e Scalable Service-Oriented Middleware over IP (SOME/IP).

Note

AWSL' IoT FleetWise supporta il middleware ROS 2 per i dati dei sistemi di visione.

Segnale del decoder

Fornisce informazioni di decodifica dettagliate per un segnale specifico. Ogni segnale specificato nel modello di veicolo deve essere associato a un segnale di decodifica. Se il manifesto del decodificatore contiene interfacce di rete CAN, deve contenere segnali di decodifica CAN. Se il manifesto del decodificatore contiene interfacce di rete OBD, deve contenere segnali di decodifica OBD.

Il manifesto del decodificatore deve contenere segnali di decodifica dei messaggi se contiene anche interfacce middleware del veicolo.

Veicolo

Una rappresentazione virtuale del veicolo fisico, ad esempio un'auto o un camion. I veicoli sono esempi di modelli di veicoli. I veicoli creati con lo stesso modello di veicolo ereditano lo stesso gruppo di segnali. Ogni veicolo corrisponde a AWS IoT qualcosa.

Parco istanze

Una flotta rappresenta un gruppo di veicoli. Prima di poter gestire facilmente una flotta di veicoli, è necessario associare i singoli veicoli a una flotta.

Campagna

Contiene schemi di raccolta dati. Definisci una campagna nel cloud e la distribuisce su un veicolo o una flotta. Le campagne forniscono al software Edge Agent istruzioni su come selezionare, raccogliere e trasferire dati sul cloud.

Schema di raccolta dei dati

Gli schemi di raccolta dei dati forniscono al software Edge Agent istruzioni su come raccogliere i dati. Attualmente, l'AWS IoT FleetWise supporta lo schema di raccolta basato sulle condizioni e lo schema di raccolta basato sul tempo.

Schema di raccolta basato sulle condizioni

Usa un'espressione logica per riconoscere quali dati raccogliere. Il software Edge Agent raccoglie i dati quando viene soddisfatta la condizione. Ad esempio, se l'espressione è `$variable.myVehicle.InVehicleTemperature >35.0`, il software Edge Agent raccoglie valori di temperatura superiori a 35,0.

Schema di raccolta basato sul tempo

Specificate un periodo di tempo in millisecondi per definire la frequenza di raccolta dei dati. Ad esempio, se il periodo di tempo è 10.000 millisecondi, il software Edge Agent raccoglie i dati una volta ogni 10 secondi.

Caratteristiche dell'AWSIoT FleetWise

Di seguito sono elencate le caratteristiche principali dell'AWSIoT FleetWise.

Modellazione di veicoli

Crea rappresentazioni virtuali dei tuoi veicoli e applica un formato comune per organizzare i segnali dei veicoli. AWS L'IoT FleetWise supporta la [Vehicle Signal Specification \(VSS\)](#) che puoi utilizzare per standardizzare i segnali dei veicoli.

Raccolta di dati basata su schemi

Definisci schemi per trasferire sul cloud solo i dati di alto valore dei veicoli. È possibile definire schemi basati sulle condizioni per controllare quali dati raccogliere, ad esempio valori di temperatura a bordo del veicolo superiori a 40 gradi. È inoltre possibile definire schemi basati sul tempo per controllare la frequenza di raccolta dei dati.

Edge Agent per il FleetWise software AWS IoT

Il software Edge Agent in esecuzione sui veicoli facilita la comunicazione tra i veicoli e il cloud. Mentre i veicoli sono connessi al cloud, il software Edge Agent riceve continuamente schemi di raccolta dati e raccoglie i dati di conseguenza.

Servizi correlati

AWSL'IoT FleetWise si integra con i seguenti AWS servizi per migliorare la disponibilità e la scalabilità delle tue soluzioni cloud.

- AWS IoT Core— Registra e controlla AWS IoT i dispositivi che caricano i dati del veicolo su AWS IoT FleetWise. Per ulteriori informazioni, consulta [Che cos'è AWS IoT?](#) nella Guida per gli sviluppatori di AWS IoT.
- Amazon Timestream: utilizza un database di serie temporali per archiviare e analizzare i dati del veicolo. Per ulteriori informazioni, consulta [What is Amazon Timestream nella Amazon Timestream Developer Guide](#).

- Amazon S3: utilizza un servizio di storage di oggetti per archiviare e gestire i dati del veicolo. Per ulteriori informazioni, consulta [Cos'è Amazon S3](#) nella Guida per l'utente di Amazon Simple Storage Service.

Configurazione dell' AWS IoT FleetWise

Prima di utilizzare l' AWS IoT FleetWise per la prima volta, completa i passaggi nelle seguenti sezioni.

Argomenti

- [Configura il tuo Account AWS](#)
- [Nozioni di base sull'utilizzo nella console](#)
- [Configurazione delle impostazioni](#)

Configura il tuo Account AWS

Completa le seguenti attività per registrarti AWS e creare un utente amministrativo.

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

Note

Puoi utilizzare un ruolo collegato ai servizi con l'IoT AWS . FleetWise I ruoli collegati ai servizi sono predefiniti dall' AWS IoT FleetWise e includono le autorizzazioni necessarie a AWS FleetWise IoT per inviare metriche ad Amazon. CloudWatch Per ulteriori informazioni, consulta [Utilizzo di ruoli collegati ai servizi per AWS IoT FleetWise](#).

Nozioni di base sull'utilizzo nella console

Se non hai già effettuato l'accesso al tuo Account AWS, accedi, quindi apri la [FleetWise console AWS IoT](#). Per iniziare a usare l' AWS IoT FleetWise, crea un modello di veicolo. Un modello di veicolo standardizza il formato dei veicoli.

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. In Inizia con AWS IoT FleetWise, scegli Inizia.

Per ulteriori informazioni sulla creazione di un modello di veicolo, consulta [Crea un modello di veicolo \(console\)](#).

Configurazione delle impostazioni

Puoi utilizzare la FleetWise console o l'API AWS IoT per configurare le impostazioni per CloudWatch i parametri di Amazon Logs, Amazon CloudWatch Logs e crittografare i dati con un. Chiave gestita da AWS

Con le CloudWatch metriche, puoi monitorare l' AWS IoT FleetWise e altre AWS risorse. Puoi utilizzare le CloudWatch metriche per raccogliere e tenere traccia delle metriche, ad esempio per determinare se c'è un limite di servizio superato. Per ulteriori informazioni sulle CloudWatch metriche, consulta [Monitoraggio AWS dell'IoT FleetWise con Amazon CloudWatch](#)

Con CloudWatch Logs, l' AWS IoT FleetWise invia i dati di CloudWatch log a un gruppo di log, dove è possibile utilizzarli per identificare e mitigare eventuali problemi. Per ulteriori informazioni sui CloudWatch registri, consulta [Configurare la FleetWise registrazione AWS IoT](#)

Con la crittografia dei dati, l' AWS IoT FleetWise utilizza Chiavi gestite da AWS per crittografare i dati. Puoi anche scegliere di creare e gestire le chiavi con AWS KMS. Per ulteriori informazioni sulla crittografia, consulta [Crittografia dei dati](#).

Configurazione delle impostazioni (console)

Se non hai già effettuato l'accesso al tuo Account AWS, accedi, quindi apri la [FleetWiseconsole AWS IoT](#).

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. Nel riquadro di sinistra, scegli Impostazioni.
3. In Metriche, scegli Abilita. AWS L'IoT associa FleetWise automaticamente una policy CloudWatch gestita al ruolo collegato al servizio e abilita le metriche. CloudWatch
4. In Registrazione, scegli Modifica.
 - a. Nella sezione CloudWatch Registrazione, inserisci il gruppo Log.
 - b. Per salvare le modifiche, scegli Invia.
5. Nella sezione Crittografia, scegli Modifica.
 - a. Scegli il tipo di chiave che desideri utilizzare. Per ulteriori informazioni, consulta [Gestione delle chiavi](#).
 - i. Usa AWS la chiave: l' AWS IoT FleetWise possiede e gestisce la chiave.
 - ii. Scegli una AWS Key Management Service chiave diversa: AWS KMS keys la gestisci nel tuo account.
 - b. Per salvare le modifiche, scegli Invia.

Configura le impostazioni (AWS CLI)

In AWS CLI, registra l'account per configurare le impostazioni.

1. Per configurare le impostazioni, esegui il seguente comando.

```
aws iotfleetwise register-account
```

2. Per verificare le impostazioni, esegui il comando seguente per recuperare lo stato della registrazione.

Note

Il ruolo collegato al servizio viene utilizzato solo per pubblicare FleetWise metriche AWS IoT su. CloudWatch Per ulteriori informazioni, consulta [Utilizzo di ruoli collegati ai servizi per AWS IoT FleetWise](#).

```
aws iotfleetwise get-register-account-status
```

Example response

```
{
  "accountStatus": "REGISTRATION_SUCCESS",
  "creationTime": "2022-07-28T11:31:22.603000-07:00",
  "customerAccountId": "012345678912",
  "iamRegistrationResponse": {
    "errorMessage": "",
    "registrationStatus": "REGISTRATION_SUCCESS",
    "roleArn": "arn:aws:iam::012345678912:role/AWSIoT FleetwiseServiceRole"
  },
  "lastModificationTime": "2022-07-28T11:31:22.854000-07:00",
}
```

Lo stato della registrazione può essere uno dei seguenti:

- **REGISTRATION_SUCCESS**— La AWS risorsa è stata registrata correttamente.

- **REGISTRATION_PENDING**— AWS IoT FleetWise sta elaborando la richiesta di registrazione. Il completamento di questo processo richiede circa cinque minuti.
- **REGISTRATION_FAILURE**— AWS L'IoT non è in FleetWise grado di registrare la AWS risorsa. Riprova più tardi.

Guida introduttiva all' AWS IoT FleetWise

Con l' AWS IoT FleetWise, puoi raccogliere, trasformare e trasferire i dati del tuo veicolo. Usa i tutorial in questa sezione per iniziare a usare l'IoT AWS . FleetWise

Consulta i seguenti argomenti per saperne di più sull' AWS IoT FleetWise:

- [Inserimento di dati nel cloud](#)
- [Modellazione di veicoli](#)
- [Crea, fornisci e gestisci veicoli](#)
- [Crea e gestisci flotte](#)
- [Raccogli e trasferisci dati con le campagne](#)

Requisiti

Devi averne uno Account AWS per iniziare con l' AWS IoT FleetWise. Se non lo hai, consultare [Configurazione dell' AWS IoT FleetWise](#).

Usa una regione in cui l' AWS IoT FleetWise è disponibile. Per ulteriori informazioni, consulta [FleetWise Endpoint e quote AWS IoT](#). Puoi utilizzare il selettore della regione in AWS Management Console per passare a una di queste regioni.

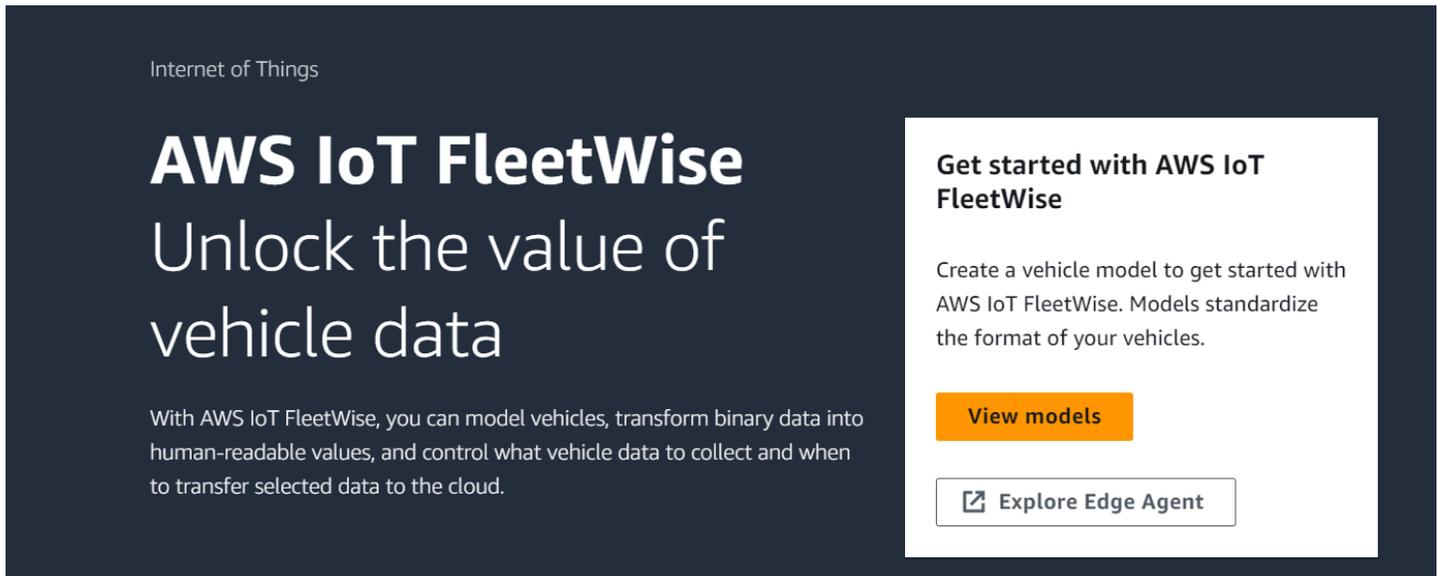
Demo del software Edge Agent

Puoi utilizzare la demo di avvio rapido di Explore Edge Agent per esplorare l' AWS IoT FleetWise e imparare a sviluppare il software Edge Agent per AWS IoT FleetWise. Questa demo utilizza un AWS CloudFormation modello. Ti guida attraverso la revisione dell'implementazione di riferimento di Edge Agent, lo sviluppo del tuo Edge Agent e quindi la distribuzione del software Edge Agent su un Amazon EC2 Graviton e la generazione di dati di esempio sul veicolo. La demo fornisce anche uno script che puoi utilizzare per creare un catalogo di segnali, un modello di veicolo, un manifesto di decodifica, un veicolo, una flotta e una campagna, il tutto nel cloud. Per ulteriori informazioni sulla demo di avvio rapido, procedi come segue per scaricare la Guida per gli sviluppatori del software Edge Agent.

Per scaricare la demo di avvio rapido

1. Passa alla [FleetWiseconsole AWS IoT](#).

2. Nella home page del servizio, nella FleetWise sezione Guida introduttiva all' AWS IoT, scegli Explore Edge Agent.



Internet of Things

AWS IoT FleetWise

Unlock the value of vehicle data

With AWS IoT FleetWise, you can model vehicles, transform binary data into human-readable values, and control what vehicle data to collect and when to transfer selected data to the cloud.

Get started with AWS IoT FleetWise

Create a vehicle model to get started with AWS IoT FleetWise. Models standardize the format of your vehicles.

[View models](#)

[Explore Edge Agent](#)

Tutorial: Guida introduttiva all' AWS IoT FleetWise (console)

Usa l' AWS IoT FleetWise per raccogliere, trasformare e trasferire l'esclusivo formato di dati dai veicoli automatizzati al cloud quasi in tempo reale. Hai accesso a informazioni dettagliate sull'intero parco veicoli. Questo può aiutarvi a rilevare e mitigare in modo efficiente i problemi di salute dei veicoli, trasferire segnali di dati di alto valore e diagnosticare i problemi da remoto, il tutto riducendo al contempo i costi.

Questo tutorial mostra come iniziare a usare l' AWS IoT FleetWise. Imparerai come creare un modello di veicolo (model manifest), un manifesto di decodifica, un veicolo e una campagna.

Per ulteriori informazioni sui componenti e i concetti chiave dell' AWS IoT FleetWise, vedere [Come FleetWise funziona AWS l'IoT](#).

Tempo stimato: circa 45 minuti.

Important

Ti verranno addebitate le FleetWise risorse AWS IoT create e utilizzate in questa demo. Per ulteriori informazioni, consulta [AWS IoT FleetWise](#) nella pagina FleetWise dei prezzi AWS IoT.

Argomenti

- [Prerequisiti](#)
- [Passaggio 1: configurare il software Edge Agent per AWS IoT FleetWise](#)
- [Fase 2: Creare un modello di veicolo](#)
- [Fase 3: Creare un manifesto del decodificatore](#)
- [Passaggio 4: Configurare un manifesto del decodificatore](#)
- [Fase 5: Creare un veicolo](#)
- [Passaggio 6: crea una campagna](#)
- [Fase 7: pulire](#)
- [Passaggi successivi](#)

Prerequisiti

Per completare questo tutorial introduttivo, devi prima disporre di quanto segue:

- Un Account AWS. Se non ne hai uno Account AWS, consulta [Creazione di un Account AWS](#) file nella Guida AWS Account Management di riferimento.
- Accesso a qualsiasi dispositivo Regione AWS che supporti l' AWS IoT FleetWise. Attualmente, l' AWS IoT FleetWise è supportato negli Stati Uniti orientali (Virginia settentrionale) e in Europa (Francoforte).
- Risorse Amazon Timestream:
 - Un database Amazon Timestream. Per ulteriori informazioni, consulta [Creare un database](#) nella Amazon Timestream Developer Guide.
 - Una tabella Amazon Timestream creata in Amazon Timestream che conterrà i tuoi dati. Per ulteriori informazioni, consulta [Creare una tabella](#) nella Amazon Timestream Developer Guide.

Passaggio 1: configurare il software Edge Agent per AWS IoT FleetWise

Note

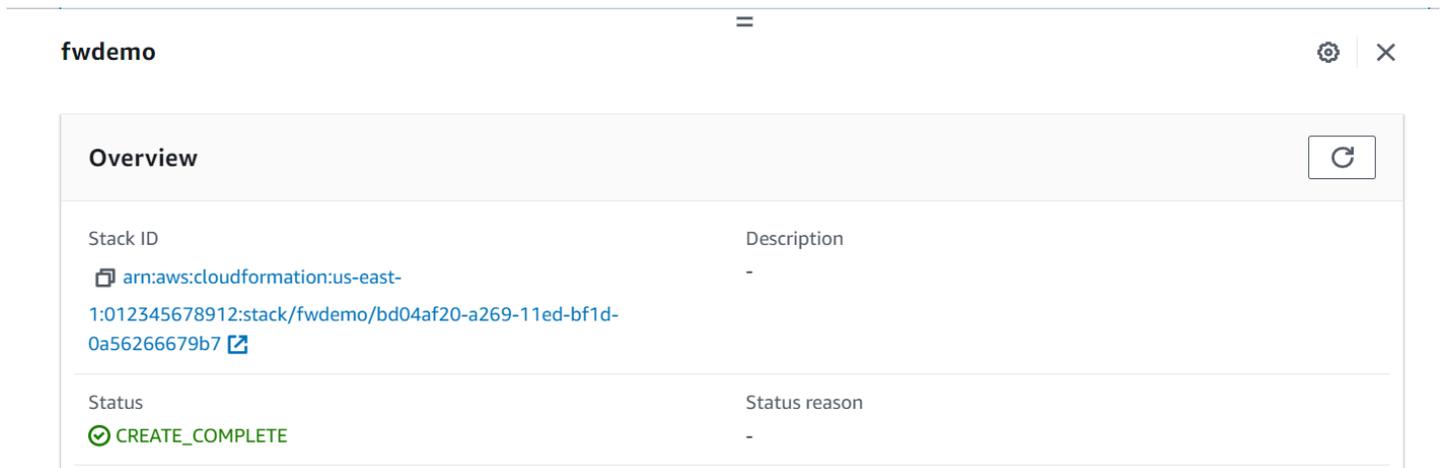
Lo CloudFormation stack in questo passaggio utilizza dati di telemetria. È inoltre possibile creare uno CloudFormation stack utilizzando i dati del sistema di visione. Per ulteriori informazioni, consulta la [Vision System Data Developer Guide](#).

I dati del sistema Vision sono in versione di anteprima e sono soggetti a modifiche.

Il tuo software Edge Agent per AWS IoT FleetWise facilita la comunicazione tra i veicoli e il cloud. Riceve istruzioni dagli schemi di raccolta dati su come raccogliere dati dai veicoli connessi al cloud.

Per configurare il software Edge Agent, in Informazioni generali, procedi come segue:

1. Apri il [CloudFormation modello di avvio](#).
2. Nella pagina Quick create stack, per Stack name, inserisci il nome del tuo stack di risorse IoT AWS . FleetWise Uno stack è un nome descrittivo che appare come prefisso sui nomi delle risorse create da questo modello. AWS CloudFormation
3. In Parametri, inserisci i valori personalizzati per i parametri relativi allo stack.
 - a. Fleetsize - Puoi aumentare il numero di veicoli della tua flotta aggiornando il parametro Fleetsize.
 - b. IoT CoreRegion - È possibile specificare la regione in cui viene creato l' AWS IoT oggetto aggiornando il CoreRegion parametro IoT. È necessario utilizzare la stessa regione utilizzata per creare i FleetWise veicoli AWS IoT. Per ulteriori informazioni Regioni AWS, consulta [Regioni e zone - Amazon Elastic Compute Cloud](#).
4. Nella sezione Capacità, seleziona la casella per confermare che AWS CloudFormation crea risorse IAM.
5. Scegli Crea stack, quindi attendi circa 15 minuti affinché lo stato dello stack visualizzi CREATE_COMPLETE.
6. Per confermare che lo stack è stato creato, scegli la scheda Informazioni sullo stack, aggiorna la vista e cerca CREATE_COMPLETE.



The screenshot shows the AWS IoT FleetWise console interface. At the top, the title 'fwdemo' is displayed. Below it, there is a table with two columns: 'Stack ID' and 'Description'. The 'Stack ID' row contains the value 'arn:aws:cloudformation:us-east-1:012345678912:stack/fwdemo/bd04af20-a269-11ed-bf1d-0a56266679b7' with a copy icon and a link icon. The 'Description' row contains a hyphen '-'. Below this, there is another table with two columns: 'Status' and 'Status reason'. The 'Status' row contains 'CREATE_COMPLETE' with a green checkmark icon. The 'Status reason' row contains a hyphen '-'. There is a refresh icon in the top right corner of the table area.

Important

Ti verranno addebitate le FleetWise risorse AWS IoT create e utilizzate in questa demo. Per ulteriori informazioni, consulta [AWS IoT FleetWise](#) nella pagina FleetWise dei prezzi AWS IoT.

Fase 2: Creare un modello di veicolo

Important

Non è possibile creare un modello di veicolo con segnali di dati del sistema di visione nella FleetWise console AWS IoT. Utilizza invece il AWS CLI.

Utilizzi i modelli di veicoli per standardizzare il formato dei veicoli, e per contribuire a definire la relazione tra i segnali nei veicoli che crei. Quando si crea un modello di veicolo, viene creato anche un catalogo di segnali. Un catalogo di segnali è una raccolta di segnali standardizzati che possono essere riutilizzati per creare modelli di veicoli. I segnali sono strutture fondamentali che definisci per contenere i dati del veicolo e i relativi metadati. Al momento, il FleetWise servizio AWS IoT supporta solo un catalogo di segnali Regione AWS per account. Questo aiuta a verificare che i dati elaborati da una flotta di veicoli siano coerenti.

Per creare un modello di veicolo

1. Apri la FleetWise console AWS IoT.
2. Nel pannello di navigazione, scegli Modelli di veicoli.

3. Nella pagina Modelli di veicolo, scegli Crea modello di veicolo.
4. Nella sezione Informazioni generali, inserisci il nome del tuo modello di veicolo, ad esempio Vehicle1, e una descrizione opzionale. Quindi scegli Successivo.
5. Scegli uno o più segnali dal catalogo dei segnali. Puoi filtrare i segnali per nome nel catalogo di ricerca o sceglierli dall'elenco. Ad esempio, puoi scegliere i segnali per la pressione dei pneumatici e la pressione dei freni in modo da poter raccogliere i dati relativi a questi segnali. Seleziona Successivo.
6. Scegli i tuoi file.dbc e caricali dal tuo dispositivo locale. Seleziona Successivo.

 Note

Per questo tutorial, puoi scaricare un [file.dbc di esempio da caricare](#) per questa fase.

7. Aggiungi gli attributi al modello del veicolo, quindi scegli Avanti.
 - a. Nome - Inserisci il nome dell'attributo del veicolo, ad esempio il nome del produttore o la data di produzione.
 - b. Tipo di dati - Nel menu Tipo di dati, scegli un tipo di dati.
 - c. Unità - (Facoltativo) Immettere un valore unitario, ad esempio chilometro o gradi Celsius.
 - d. Percorso - (facoltativo) Immettete un nome per il percorso di un segnale, ad esempio Vehicle.Engine.Light. Il punto (.) indica che si tratta di un segnale secondario.
 - e. Valore predefinito - (Facoltativo) Inserire un valore predefinito.
 - f. Descrizione - (Facoltativo) Inserire una descrizione dell'attributo.
8. Rivedi le tue configurazioni. Al termine, scegli Crea. Viene visualizzata una notifica che indica che il modello di veicolo è stato creato con successo.

✔ **Vehicle model created**
You successfully created the vehicle model: demo. ✕

AWS IoT FleetWise > Vehicle models > Demo

demo

[Duplicate](#) [Create vehicle](#) [Create decoder manifest](#)

When a decoder manifest is associated with a vehicle model, you can create a vehicle. To use the API to create vehicles with this vehicle model, follow the instructions in the AWS IoT FleetWise Developer Guide. After you create vehicles, you can create campaigns for them.

Summary [Info](#)

Vehicle model ARN arn:aws:iotfleetwise:us-east-1:012345678912:model-manifest/demo	Status ✔ ACTIVE	Date created February 01, 2023 at 14:40 (UTC-05)
Signal catalog ARN arn:aws:iotfleetwise:us-east-1:012345678912:signal-catalog/DefaultSignalCatalog	Description -	Last modified February 01, 2023 at 14:40 (UTC-05)

Fase 3: Creare un manifesto del decodificatore

I manifesti del decoder sono associati ai modelli di veicoli che crei. Contengono informazioni che aiutano l'AWS IoT a FleetWise decodificare e trasformare i dati dei veicoli da un formato binario a valori leggibili dall'uomo che possono essere analizzati. Le interfacce di rete e i segnali del decodificatore sono componenti che aiutano a configurare i manifesti del decoder. Un'interfaccia di rete contiene informazioni sul protocollo CAN o OBD utilizzato dalla rete del veicolo. Il segnale del decodificatore fornisce informazioni di decodifica per un segnale specifico.

Per creare un manifesto del decodificatore

1. Apri la FleetWise console AWS IoT.
2. Nel pannello di navigazione, scegli Modelli di veicoli.
3. Nella sezione Modelli di veicoli, scegli il modello di veicolo che desideri utilizzare per creare un manifesto di decodifica.
4. Scegli Crea manifesto del decoder.

Passaggio 4: Configurare un manifesto del decodificatore

Per configurare un manifesto del decodificatore

Important

Non è possibile configurare i segnali dei dati del sistema di visione nei manifesti del decoder utilizzando la console AWS FleetWise IoT. Utilizza invece il. AWS CLI Per ulteriori informazioni, consulta [Create un manifesto del decodificatore \(AWS CLI\)](#).

1. Per aiutarvi a identificare il manifesto del decoder, inserite un nome e una descrizione opzionale. Quindi, seleziona Next (Successivo).
2. Per aggiungere una o più interfacce di rete, scegliete il tipo CAN_INTERFACE o OBD_INTERFACE.
 - Interfaccia diagnostica di bordo (OBD) - Scegliete questo tipo di interfaccia se desiderate un protocollo che definisca il modo in cui i dati di autodiagnostica vengono comunicati tra le unità di controllo elettroniche (ECU). Questo protocollo fornisce una serie di codici diagnostici di errore (DTC) standard che possono aiutarvi a risolvere i problemi del veicolo.
 - Interfaccia Controller Area Network (CAN bus) - Scegliete questo tipo di interfaccia se desiderate un protocollo che definisca il modo in cui i dati vengono comunicati tra le ECU. Le ECU possono essere unità di controllo del motore, airbag o sistema audio.
3. Immettete il nome di un'interfaccia di rete.
4. Per aggiungere segnali all'interfaccia di rete, scegliete uno o più segnali dall'elenco.
5. Scegliete un segnale di decodifica per il segnale aggiunto nel passaggio precedente. Per fornire informazioni di decodifica, carica un file.dbc. Ogni segnale del modello di veicolo deve essere associato a un segnale di decodifica che puoi scegliere dall'elenco.
6. Per aggiungere un'altra interfaccia di rete, scegli Aggiungi interfaccia di rete. Quando hai finito di aggiungere interfacce di rete, scegli Avanti.
7. Controlla le configurazioni, quindi scegli Crea. Viene visualizzata una notifica che indica che il manifesto del decoder è stato creato con successo.

Fase 5: Creare un veicolo

Nell' AWS IoT FleetWise, i veicoli sono rappresentazioni virtuali del veicolo fisico reale. Tutti i veicoli creati con lo stesso modello di veicolo ereditano lo stesso gruppo di segnali e ogni veicolo creato corrisponde a un oggetto IoT appena creato. È necessario associare tutti i veicoli a un manifesto di decodifica.

Prerequisiti

1. Verifica di aver già creato il modello del veicolo e il manifesto del decodificatore. Inoltre, verifica che lo stato del modello di veicolo sia **ATTIVO**.
 - a. Per verificare che lo stato del modello di veicolo sia **ATTIVO**, apri la FleetWise console AWS IoT.
 - b. Nel pannello di navigazione, scegli Modelli di veicoli.
 - c. Nella sezione Riepilogo, sotto Stato, controlla lo stato del veicolo.

✔ **Vehicle model created**
You successfully created the vehicle model: demo.

AWS IoT FleetWise > Vehicle models > Demo

demo

[Duplicate](#) [Create vehicle](#) [Create decoder manifest](#)

When a decoder manifest is associated with a vehicle model, you can create a vehicle. To use the API to create vehicles with this vehicle model, follow the instructions in the AWS IoT FleetWise Developer Guide. After you create vehicles, you can create campaigns for them.

Summary [Info](#)

Vehicle model ARN arn:aws:iotfleetwise:us-east-1:012345678912:model-manifest/demo	Status ACTIVE	Date created February 01, 2023 at 14:40 (UTC-05)
Signal catalog ARN arn:aws:iotfleetwise:us-east-1:012345678912:signal-catalog/DefaultSignalCatalog	Description -	Last modified February 01, 2023 at 14:40 (UTC-05)

Per creare un veicolo

1. Apri la FleetWise console AWS.
2. Nel pannello di navigazione, scegli Veicoli.

3. Scegli Crea veicolo.
4. Per definire le proprietà del veicolo, inserisci il nome del veicolo, quindi scegli un manifesto del modello (modello del veicolo) e un manifesto del decodificatore.
5. (Facoltativo) Per definire gli attributi del veicolo, inserite una coppia chiave-valore, quindi scegliete Aggiungi attributi.
6. (Facoltativo) Per etichettare la tua risorsa AWS, aggiungi i tag, quindi scegli Aggiungi nuovo tag.
7. Seleziona Successivo.
8. Per configurare il certificato del veicolo, puoi caricare il tuo certificato o scegliere Genera automaticamente un nuovo certificato. Ti consigliamo di generare automaticamente il certificato per una configurazione più rapida. Se disponi già di un certificato, puoi scegliere di utilizzarlo al suo posto.
9. Scarica i file delle chiavi pubbliche e private, quindi scegli Avanti.
10. Per allegare una politica al certificato del veicolo, puoi inserire il nome di una polizza esistente o crearne una nuova. Per creare una nuova politica, scegli Crea politica, quindi scegli Avanti.
11. Rivedi le tue configurazioni. Quando hai finito, scegli Crea veicolo.

Passaggio 6: crea una campagna

Nell' AWS IoT FleetWise, le campagne vengono utilizzate per facilitare la selezione, la raccolta e il trasferimento di dati dai veicoli al cloud. Le campagne contengono schemi di raccolta dati che forniscono al software Edge Agent istruzioni su come raccogliere dati con uno schema di raccolta basato sulle condizioni o uno schema di raccolta basato sul tempo.

Per creare una campagna

1. Apri la FleetWise console AWS IoT.
2. Nel riquadro di navigazione, scegli Campagne.
3. Scegliere Create campaign (Crea campagna).
4. Inserisci il nome della campagna e una descrizione opzionale.
5. Per configurare lo schema di raccolta dati della tua campagna, puoi definire manualmente lo schema di raccolta dei dati o caricare un file.json dal tuo dispositivo locale. Il caricamento di un file.json definisce automaticamente lo schema di raccolta dei dati.
 - a. Per definire manualmente lo schema di raccolta dei dati, scegli Definisci schema di raccolta dati e scegli il tipo di schema di raccolta dati che desideri utilizzare per la tua campagna.

- Puoi scegliere uno schema di raccolta basato sulle condizioni o uno schema di raccolta basato sul tempo.
- b. Se scegli uno schema di raccolta basato sul tempo, devi specificare per quanto tempo la campagna raccoglierà i dati del veicolo.
 - c. Se scegli uno schema di raccolta basato sulle condizioni, devi specificare un'espressione per riconoscere quali dati raccogliere. Assicuratevi di specificare il nome del segnale come variabile, operatore di confronto e valore di confronto.
 - d. (Facoltativo) Scegliete la versione linguistica dell'espressione o mantenetele come valore predefinito di 1.
 - e. (Facoltativo) Specificate l'intervallo di attivazione tra due eventi di raccolta dati.
 - f. Per raccogliere dati, scegli la condizione della modalità Trigger per il software Edge Agent. Per impostazione predefinita, il FleetWise software Edge Agent for AWS IoT raccoglie sempre i dati ogni volta che viene soddisfatta la condizione. In alternativa, può raccogliere dati solo quando la condizione viene soddisfatta per la prima volta, al primo trigger.
 - g. (Facoltativo) È possibile scegliere opzioni di schema più avanzate.
6. Per specificare i segnali da cui lo schema di raccolta dati raccoglierà i dati, cercate il nome del segnale dal menu.
 7. (Facoltativo) È possibile scegliere un numero massimo di campioni o un intervallo di campionamento minimo. Puoi anche aggiungere altri segnali.
 8. Seleziona Successivo.
 9. Definisci la destinazione di archiviazione in cui desideri che la campagna trasferisca i dati. Puoi archiviare dati in Amazon S3 o Amazon Timestream.
 - a. Amazon S3: scegli il bucket S3 con le autorizzazioni per farlo AWS IoT FleetWise .
 - b. Amazon Timestream: scegli il nome del database e della tabella Timestream. Inserisci un ruolo IAM che AWS IoT FleetWise consenta di inviare dati a Timestream.
 10. Seleziona Successivo.
 11. Scegli gli attributi o i nomi dei veicoli dalla casella di ricerca.
 12. Inserisci il valore relativo all'attributo o al nome che hai scelto per il veicolo.
 13. Scegli i veicoli da cui la tua campagna raccoglierà i dati. Quindi, seleziona Next (Successivo).
 14. Controlla le configurazioni della tua campagna, quindi scegli Crea campagna. Tu o il tuo team dovete distribuire la campagna sui veicoli.

Fase 7: pulire

Per evitare ulteriori addebiti per le risorse utilizzate durante questo tutorial, elimina lo AWS CloudFormation stack e tutte le risorse dello stack.

Per eliminare lo stack AWS CloudFormation

1. Apri la [AWS CloudFormation console](#).
2. Dall'elenco degli stack, scegli lo stack che hai creato nel passaggio 1.
3. Scegli Elimina.
4. Per confermare l'eliminazione, scegliere Delete (Elimina). L'eliminazione dello stack richiede circa 15 minuti.

Passaggi successivi

1. Puoi elaborare e visualizzare i dati sui veicoli raccolti dalla tua campagna. Per ulteriori informazioni, consulta [Elaborazione e visualizzazione dei dati del veicolo](#).
2. Puoi risolvere e risolvere i problemi con l'IoT AWS . FleetWise Per ulteriori informazioni, consulta [Risoluzione dei problemi AWS IoT FleetWise](#).

Inserimento di dati nel cloud

Il FleetWise software Edge Agent for AWS IoT, se installato e funzionante nei veicoli, è progettato per facilitare la comunicazione sicura tra i veicoli e il cloud.

Note

- AWS FleetWise L'IoT non è destinato all'uso o in associazione con il funzionamento di ambienti pericolosi o sistemi critici che possono causare gravi lesioni personali o morte o causare danni ambientali o materiali. I dati sui veicoli raccolti tramite l'uso dell'AWSIoT sono solo a scopo informativo e non FleetWise è possibile utilizzare l'AWSIoT FleetWise per controllare o gestire le funzioni del veicolo.
- I dati del veicolo raccolti tramite l'uso dell'AWSIoT FleetWise devono essere valutati per verificarne l'accuratezza in base al caso d'uso, anche allo scopo di soddisfare eventuali obblighi di conformità previsti dalle normative sulla sicurezza dei veicoli applicabili (come gli obblighi di monitoraggio e segnalazione della sicurezza). Tale valutazione dovrebbe includere la raccolta e la revisione delle informazioni attraverso altri strumenti e fonti standard del settore (come i rapporti dei conducenti di veicoli).

Per importare dati nel cloud, procedi come segue:

1. Sviluppa e installa il FleetWise software Edge Agent for AWS IoT nel tuo veicolo. Per ulteriori informazioni su come utilizzare il software Edge Agent, procedi come segue per scaricare la [Guida per gli sviluppatori del FleetWise software Edge Agent for AWS IoT](#).
 1. Passa alla [FleetWise console AWS IoT](#).
 2. Nella home page del servizio, nella FleetWise sezione Introduzione all'AWSIoT, scegli Explore Edge Agent.
2. Crea o importa un catalogo di segnali contenente segnali che utilizzerai per creare un modello di veicolo. Per ulteriori informazioni, consultare [Crea un catalogo di segnali \(AWS CLI\)](#) e [Importa un catalogo di segnali \(AWS CLI\)](#).

Note

- Se utilizzi la FleetWise console AWS IoT per creare il primo modello di veicolo, non è necessario creare manualmente un catalogo di segnali. Quando crei il tuo primo modello di veicolo, l'AWS IoT crea FleetWise automaticamente un catalogo di segnali per te. Per ulteriori informazioni, consulta [Crea un modello di veicolo \(console\)](#).
- AWS FleetWise Attualmente IoT supporta un catalogo di segnali per ogni AWS account Regione AWS.

3. Usa i segnali nel catalogo dei segnali per creare un modello di veicolo. Per ulteriori informazioni, consulta [Crea un modello di veicolo](#).

Note

- Se utilizzi la FleetWise console AWS IoT per creare un modello di veicolo, puoi caricare file.dbc per importare segnali. .dbc è un formato di file supportato dai database Controller Area Network (CAN bus). Dopo la creazione del modello del veicolo, i nuovi segnali vengono aggiunti automaticamente al catalogo dei segnali. Per ulteriori informazioni, consulta [Crea un modello di veicolo \(console\)](#).
- Se si utilizza l'operazione `CreateModelManifest` API per creare un modello di veicolo, è necessario utilizzare l'operazione `UpdateModelManifest` API per attivare il modello di veicolo. Per ulteriori informazioni, consulta [Aggiornare un modello di veicolo \(AWS CLI\)](#).
- Se utilizzi la FleetWise console AWS IoT per creare un modello di veicolo, AWS IoT attiva FleetWise automaticamente il modello di veicolo per te.

4. Crea un manifesto del decodificatore. Il manifesto del decodificatore contiene informazioni di decodifica per ogni segnale specificato nel modello di veicolo creato nel passaggio precedente. Il manifesto del decodificatore è associato al modello di veicolo che hai creato. Per ulteriori informazioni, consulta [Crea e gestisci i manifesti del decodificatore](#).

Note

- Se si utilizza l'operazione `CreateDecoderManifest` API per creare un manifesto del decodificatore, è necessario utilizzare l'operazione `UpdateDecoderManifest` API per

attivare il manifesto del decodificatore. Per ulteriori informazioni, consulta [Aggiornare un manifesto del decoder \(AWS CLI\)](#).

- Se utilizzi la FleetWise console AWS IoT per creare un manifesto del decoder, AWS IoT attiva FleetWise automaticamente il manifesto del decoder per te.

5. Crea veicoli dal modello del veicolo. I veicoli creati dallo stesso modello di veicolo ereditano lo stesso gruppo di segnali. È necessario AWS IoT Core eseguire il provisioning del veicolo prima di poter importare dati nel cloud. Per ulteriori informazioni, consulta [Crea, fornisci e gestisci veicoli](#).
6. (Facoltativo) Crea una flotta per rappresentare un gruppo di veicoli, quindi associa i singoli veicoli alla flotta. Questo ti aiuta a gestire più veicoli contemporaneamente. Per ulteriori informazioni, consulta [Crea e gestisci flotte](#).
7. Crea campagne. Le campagne vengono distribuite su un veicolo o su una flotta di veicoli. Le campagne forniscono al software Edge Agent istruzioni su come selezionare, raccogliere e trasferire i dati sul cloud. Per ulteriori informazioni, consulta [Raccogli e trasferisci dati con le campagne](#).

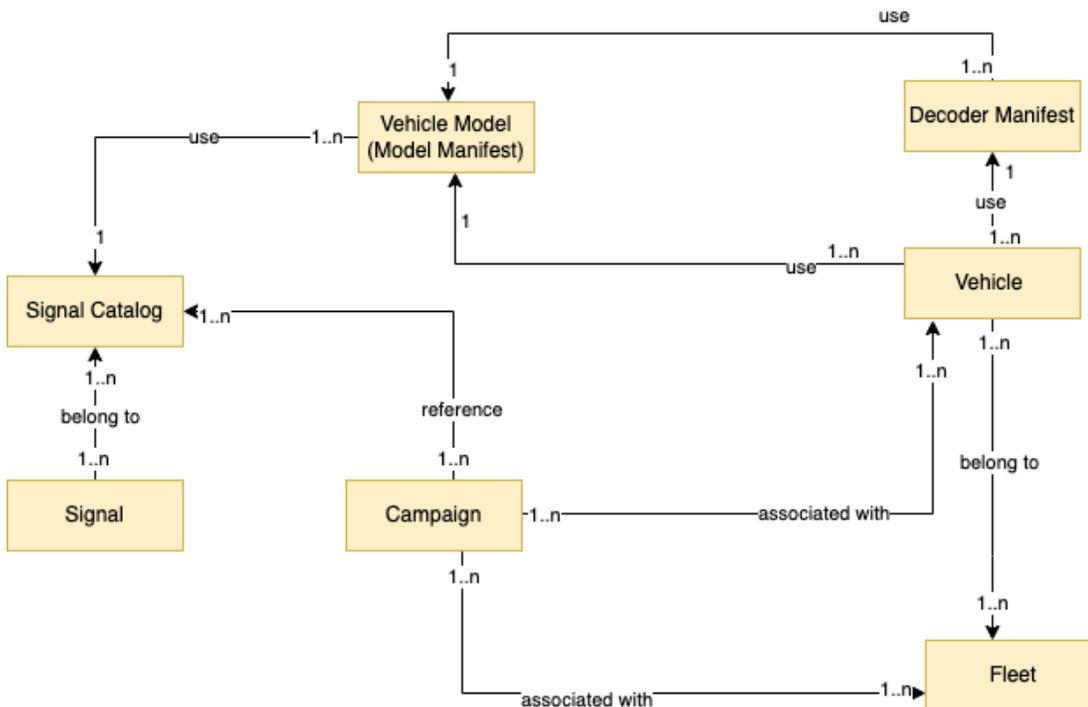
Note

È necessario utilizzare l'operazione UpdateCampaign API per approvare la campagna prima che l'AWS IoT FleetWise possa implementarla sul veicolo o sulla flotta. Per ulteriori informazioni, consulta [Aggiorna una campagna \(AWS CLI\)](#).

Il software Edge Agent trasferisce i dati del veicolo AWS IoT Core utilizzando l'argomento riservato `$aws/iotfleetwise/vehicles/vehicleName/signals`, che invia i dati all'AWS IoT FleetWise. AWS IoT fornisce FleetWise quindi i dati a una tabella Timestream o a un bucket Amazon S3. Puoi usare Timestream per interrogare i tuoi dati e usare Amazon o QuickSight Grafana per visualizzarli. Per ulteriori informazioni, consulta [Elaborazione e visualizzazione dei dati del veicolo](#).

Modellazione di veicoli

AWS FleetWise L'IoT fornisce un framework di modellazione dei veicoli che puoi utilizzare per creare rappresentazioni virtuali dei tuoi veicoli nel cloud. Segnali, cataloghi di segnali, modelli di veicoli e manifesti di decodifica sono i componenti principali con cui lavori per modellare i tuoi veicoli.



Signal

I segnali sono strutture fondamentali che definisci per contenere i dati del veicolo e i relativi metadati. Un segnale può essere un attributo, un ramo, un sensore o un attuatore. Ad esempio, è possibile creare un sensore per ricevere i valori di temperatura all'interno del veicolo e memorizzarne i metadati, tra cui il nome del sensore, un tipo di dati e un'unità. Per ulteriori informazioni, consulta [Crea e gestisci cataloghi di segnali](#).

Catalogo dei segnali

Un catalogo di segnali contiene una raccolta di segnali. I segnali presenti in un catalogo di segnali possono essere utilizzati per modellare veicoli che utilizzano protocolli e formati di dati diversi. Ad esempio, ci sono due auto prodotte da case automobilistiche diverse: una utilizza il protocollo Control Area Network (CAN bus); l'altra utilizza il protocollo OBD (On-board Diagnostics). È possibile definire un sensore nel catalogo dei segnali per ricevere i valori di temperatura all'interno del veicolo. Questo sensore può essere utilizzato per rappresentare le termocoppie di entrambe le auto. Per ulteriori informazioni, consulta [Crea e gestisci cataloghi di segnali](#).

Modello di veicolo (manifesto del modello)

I modelli di veicolo sono strutture dichiarative che è possibile utilizzare per standardizzare il formato dei veicoli e definire le relazioni tra i segnali nei veicoli. I modelli di veicoli applicano informazioni coerenti su più veicoli dello stesso tipo. Aggiungi segnali per creare modelli di veicoli. Per ulteriori informazioni, consulta [Crea e gestisci modelli di veicoli](#).

Manifesto del decoder

I manifesti del decoder contengono informazioni di decodifica per ogni segnale nei modelli di veicoli. I sensori e gli attuatori dei veicoli trasmettono messaggi di basso livello (dati binari). Con i decoder manifest, l' AWS IoT FleetWise è in grado di trasformare i dati binari in valori leggibili dall'uomo. Ogni manifesto del decoder è associato a un modello di veicolo. Per ulteriori informazioni, consulta [Crea e gestisci i manifesti del decodificatore](#).

Puoi utilizzare la FleetWise console o l'API AWS IoT per modellare i veicoli nel modo seguente.

1. Crea o importa un catalogo di segnali contenente segnali che utilizzerai per creare un modello di veicolo. Per ulteriori informazioni, consultare [Crea un catalogo di segnali \(AWS CLI\)](#) e [Importa un catalogo di segnali \(AWS CLI\)](#).

Note

- Se utilizzi la FleetWise console AWS IoT per creare il primo modello di veicolo, non è necessario creare manualmente un catalogo di segnali. Quando crei il tuo primo modello di veicolo, l' AWS IoT crea FleetWise automaticamente un catalogo di segnali per te. Per ulteriori informazioni, consulta [Crea un modello di veicolo \(console\)](#).
- AWS FleetWise Attualmente IoT supporta un catalogo di segnali per ogni AWS account Regione AWS.

2. Usa i segnali nel catalogo dei segnali per creare un modello di veicolo. Per ulteriori informazioni, consulta [Crea un modello di veicolo](#).

Note

- Se utilizzi la FleetWise console AWS IoT per creare un modello di veicolo, puoi caricare file.dbc per importare segnali. .dbc è un formato di file supportato dai database Controller Area Network (CAN bus). Dopo la creazione del modello del veicolo, i

nuovi segnali vengono aggiunti automaticamente al catalogo dei segnali. Per ulteriori informazioni, consulta [Crea un modello di veicolo \(console\)](#).

- Se si utilizza l'operazione `CreateModelManifest` API per creare un modello di veicolo, è necessario utilizzare l'operazione `UpdateModelManifest` API per attivare il modello di veicolo. Per ulteriori informazioni, consulta [Aggiornare un modello di veicolo \(AWS CLI\)](#).
- Se utilizzi la FleetWise console AWS IoT per creare un modello di veicolo, AWS IoT attiva FleetWise automaticamente il modello di veicolo per te.

3. Crea un manifesto del decodificatore. Il manifesto del decodificatore contiene informazioni di decodifica per ogni segnale specificato nel modello di veicolo creato nel passaggio precedente. Il manifesto del decodificatore è associato al modello di veicolo che hai creato. Per ulteriori informazioni, consulta [Crea e gestisci i manifesti del decodificatore](#).

Note

- Se si utilizza l'operazione `CreateDecoderManifest` API per creare un manifesto del decodificatore, è necessario utilizzare l'operazione `UpdateDecoderManifest` API per attivare il manifesto del decodificatore. Per ulteriori informazioni, consulta [Aggiornare un manifesto del decoder \(AWS CLI\)](#).
- Se utilizzi la FleetWise console AWS IoT per creare un manifesto del decoder, AWS IoT attiva FleetWise automaticamente il manifesto del decoder per te.

I database CAN bus supportano il formato di file.dbc. È possibile caricare file.dbc per importare segnali e segnali di decodifica. Per ottenere un file.dbc di esempio, procedi come segue.

Per ottenere un file.dbc

1. [Scarica il file.zip. EngineSignals](#)
2. Passare alla directory in cui è stato scaricato il file `EngineSignals.zip`.
3. Decomprimi il file e salvalo localmente come `EngineSignals.dbc`

Argomenti

- [Crea e gestisci cataloghi di segnali](#)
- [Crea e gestisci modelli di veicoli](#)

- [Crea e gestisci i manifesti del decodificatore](#)

Crea e gestisci cataloghi di segnali

Note

È possibile scaricare uno [script dimostrativo](#) per convertire i messaggi ROS 2 in file VSS JSON compatibili con il catalogo dei segnali. Per ulteriori informazioni, consulta la [Vision System Data Developer Guide](#).

Un catalogo di segnali è una raccolta di segnali standardizzati che possono essere riutilizzati per creare modelli di veicoli. AWS L'IoT FleetWise supporta la [Vehicle Signal Specification \(VSS\)](#) che puoi seguire per definire i segnali. Un segnale può essere uno dei seguenti tipi.

Attributo

Gli attributi rappresentano informazioni statiche che generalmente non cambiano, come il produttore e la data di produzione.

Ramo

I rami rappresentano i segnali in una struttura annidata. I rami mostrano le gerarchie dei segnali. Ad esempio, il `Vehicle` ramo ha un ramo figlio, `Powertrain`. Il `Powertrain` ramo ha un ramo figlio, `combustionEngine`. Per localizzare il `combustionEngine` ramo, usa l'espressione `Vehicle.Powertrain.combustionEngine`.

Sensor

I dati del sensore riportano lo stato attuale del veicolo e le variazioni nel tempo, al variare dello stato del veicolo, ad esempio i livelli del fluido, le temperature, le vibrazioni o la tensione.

Actuator

I dati degli attuatori riportano lo stato di un dispositivo del veicolo, come motori, riscaldatori e serrature delle porte. La modifica dello stato di un dispositivo del veicolo può aggiornare i dati dell'attuatore. Ad esempio, è possibile definire un attuatore per rappresentare il riscaldatore. L'attuatore riceve nuovi dati quando si accende o si spegne il riscaldatore.

Struttura personalizzata

Una struttura personalizzata (nota anche come struttura) rappresenta una struttura di dati complessa o di ordine superiore. Facilita l'associazione o il raggruppamento logico di dati provenienti dalla stessa fonte. Una struttura viene utilizzata quando i dati vengono letti o scritti in un'operazione atomica, ad esempio per rappresentare un tipo di dati complesso o una forma di ordine superiore.

Un segnale di tipo di struttura viene definito nel catalogo dei segnali utilizzando un riferimento a un tipo di dati di struttura anziché a un tipo di dati primitivo. Le strutture possono essere utilizzate per tutti i tipi di segnali, inclusi sensori, attributi, attuatori e tipi di dati dei sistemi di visione. Se viene inviato o ricevuto un segnale di tipo struttura, l' AWS IoT FleetWise si aspetta che tutti gli elementi inclusi abbiano valori validi, quindi tutti gli elementi sono obbligatori. Ad esempio, se una struttura contiene gli elementi `Vehicle.Camera.Image.Height`, `Vehicle.Camera.Image.Width` e `Vehicle.Camera.image.data`, è previsto che il segnale inviato contenga valori per tutti questi elementi.

Note

I dati del sistema di visione sono in versione di anteprima e sono soggetti a modifiche.

Proprietà personalizzata

Una proprietà personalizzata rappresenta un membro della struttura dati complessa. Il tipo di dati della proprietà può essere primitivo o un'altra struttura.

Quando si rappresenta una forma di ordine superiore utilizzando una struttura e una proprietà personalizzata, la forma di ordine superiore desiderata viene sempre definita e visualizzata come una struttura ad albero. La proprietà personalizzata viene utilizzata per definire tutti i nodi foglia mentre la struttura viene utilizzata per definire tutti i nodi non fogliari.

Note

- Se utilizzi la FleetWise console AWS IoT per creare il primo modello di veicolo, non è necessario creare manualmente un catalogo di segnali. Quando crei il tuo primo modello di veicolo, l' AWS IoT crea FleetWise automaticamente un catalogo di segnali per te. Per ulteriori informazioni, consulta [Crea un modello di veicolo \(console\)](#).

- Se utilizzi la FleetWise console AWS IoT per creare un modello di veicolo, puoi caricare file.dbc per importare segnali. .dbc è un formato di file supportato dai database Controller Area Network (CAN bus). Dopo la creazione del modello del veicolo, i nuovi segnali vengono aggiunti automaticamente al catalogo dei segnali. Per ulteriori informazioni, consulta [Crea un modello di veicolo \(console\)](#).
- AWS L' IoT FleetWise attualmente supporta un catalogo di segnali Account AWS per ogni regione.

AWS L' IoT FleetWise fornisce le seguenti operazioni API che è possibile utilizzare per creare e gestire cataloghi di segnali.

- [CreateSignalCatalog](#)— Crea un nuovo catalogo di segnali.
- [ImportSignalCatalog](#)— Importa segnali per creare un catalogo di segnali caricando un file JSON. I segnali devono essere definiti seguendo VSS e salvati nel formato JSON.
- [UpdateSignalCatalog](#)— Aggiorna un catalogo di segnali esistente aggiornando, rimuovendo o aggiungendo segnali.
- [DeleteSignalCatalog](#)— Elimina un catalogo di segnali esistente.
- [ListSignalCatalogs](#)— Recupera un elenco impaginato di riepiloghi di tutti i cataloghi di segnali.
- [ListSignalCatalogNodes](#)— Recupera un elenco impaginato di riepiloghi di tutti i segnali (nodi) in un determinato catalogo di segnali.
- [GetSignalCatalog](#)— Recupera informazioni su un catalogo di segnali.

Tutorial

- [Configura i segnali](#)
- [Crea un catalogo di segnali \(AWS CLI\)](#)
- [Importa un catalogo di segnali](#)
- [Aggiornare un catalogo di segnali \(AWS CLI\)](#)
- [Eliminare un catalogo di segnali \(AWS CLI\)](#)
- [Ottieni informazioni sul catalogo dei segnali \(AWS CLI\)](#)

Configura i segnali

Questa sezione mostra come configurare rami, attributi, sensori e attuatori.

Argomenti

- [Configurare le filiali](#)
- [Configura gli attributi](#)
- [Configura sensori o attuatori](#)
- [Configura tipi di dati complessi](#)

Configurare le filiali

Per configurare un ramo, specifica le seguenti informazioni.

- `fullyQualifiedName`— Il nome completo del ramo è il percorso del ramo più il nome del ramo. Utilizzate un punto (.) per fare riferimento a un ramo secondario. Ad esempio, `Vehicle.Chassis.SteeringWheel` è il nome completo del `SteeringWheel` ramo. `Vehicle.Chassis.` è il percorso verso questo ramo.

Il nome completo può contenere fino a 150 caratteri. Caratteri validi: `a—z`, `A—Z`, `0—9`, due punti (:) e trattino basso (_).

- (Facoltativo) `Description`: la descrizione del ramo.

La descrizione può contenere fino a 2048 caratteri. Caratteri validi: `a—z`, `A—Z`, `0—9`,: (due punti), _ (trattino basso) e - (trattino).

- (Facoltativo) `deprecationMessage` — Il messaggio di deprecazione per il nodo o il ramo che viene spostato o eliminato.

Il `DeprecationMessage` può contenere fino a 2048 caratteri. Caratteri validi: `a—z`, `A—Z`, `0—9`,: (due punti), _ (trattino basso) e - (trattino basso).

- (Facoltativo) `comment` — Un commento in aggiunta alla descrizione. Un commento può essere utilizzato per fornire informazioni aggiuntive sulla filiale, ad esempio la motivazione alla base della succursale o i riferimenti alle filiali correlate.

Il commento può contenere fino a 2048 caratteri. Caratteri validi: `a—z`, `A—Z`, `0—9`,: (due punti), _ (trattino basso) e - (trattino).

Configura gli attributi

Per configurare un attributo, specificare le seguenti informazioni.

- `dataType`— Il tipo di dati dell'attributo deve essere uno dei seguenti: INT8, UINT8, INT16, INT32, UINT32, INT64, UINT64, BOOLEAN, FLOAT, DOUBLE, STRING, UNIX_TIMESTAMP, INT8_ARRAY, UINT8_ARRAY, INT16_ARRAY, UINT16_ARRAY, INT32_ARRAY, INT_64_ARRAY, INT_64_ARRAY ARRAY, UINT64_ARRAY, BOOLEAN_ARRAY, FLOAT_ARRAY, DOUBLE_ARRAY, STRING_ARRAY, UNIX_TIMESTAMP_ARRAY, UNKNOWN o una struttura personalizzata definita nel ramo del tipo di dati. `fullyQualifiedName`
- `fullyQualifiedName`— Il nome completo dell'attributo è il percorso dell'attributo più il nome dell'attributo. Utilizzate un punto (.) per fare riferimento a un segnale secondario. Ad esempio, `Vehicle.Chassis.SteeringWheel.Diameter` è il nome completo dell'`Diameter` attributo. `Vehicle.Chassis.SteeringWheel.` è il percorso di questo attributo.

Il nome completo può contenere fino a 150 caratteri. Caratteri validi: a—z, A—Z, 0—9, : (due punti) e _ (trattino basso).

- (Facoltativo) `Description` — La descrizione dell'attributo.

La descrizione può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9, : (due punti), _ (trattino basso) e - (trattino).

- (Facoltativo) `unit` — L'unità scientifica per l'attributo, ad esempio km o Celsius.
- (Facoltativo) `min`: il valore minimo dell'attributo.
- (Facoltativo) `max` — Il valore massimo dell'attributo.
- (Facoltativo) `defaultValue` — Il valore predefinito dell'attributo.
- (Facoltativo) `assignedValue` — Il valore assegnato all'attributo.
- (Facoltativo) `allowedValues` — Un elenco di valori accettati dall'attributo.
- (Facoltativo) `deprecationMessage` — Il messaggio di deprecazione per il nodo o il ramo che viene spostato o eliminato.

Il `DeprecationMessage` può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9, : (due punti), _ (trattino basso) e - (trattino basso).

- (Facoltativo) `comment` — Un commento in aggiunta alla descrizione. Un commento può essere utilizzato per fornire informazioni aggiuntive sull'attributo, ad esempio la motivazione dell'attributo o i riferimenti agli attributi correlati.

Il commento può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9, : (due punti), _ (trattino basso) e - (trattino).

Configura sensori o attuatori

Per configurare un sensore o un attuatore, specificare le seguenti informazioni.

- `dataType`— Il tipo di dati del segnale deve essere uno dei seguenti: INT8, UINT8, INT16, UINT32, INT64, BOOLEAN, FLOAT, DOUBLE, STRING, UNIX_TIMESTAMP, INT8_ARRAY, UINT8_ARRAY, INT16_ARRAY, UINT16_ARRAY, INT32_ARRAY, INT_64ARRAY, INT_64_ARRAY ARRAY, UINT64_ARRAY, BOOLEAN_ARRAY, FLOAT_ARRAY, DOUBLE_ARRAY, STRING_ARRAY, UNIX_TIMESTAMP_ARRAY, UNKNOWN o una struttura personalizzata definita nel ramo del tipo di dati. `fullyQualifiedName`
- `fullyQualifiedName`— Il nome completo del segnale è il percorso del segnale più il nome del segnale. Utilizzate un punto (.) per fare riferimento a un segnale secondario. Ad esempio, `Vehicle.Chassis.SteeringWheel.HandsOff.HandsOffSteeringState` è il nome completo dell'`HandsOffSteeringState` attuatore. `Vehicle.Chassis.SteeringWheel.HandsOff` è il percorso verso questo attuatore.

Il nome completo può contenere fino a 150 caratteri. Caratteri validi: a—z, A—Z, 0—9, : (due punti) e _ (trattino basso).

- (Facoltativo) `Description` — La descrizione del segnale.

La descrizione può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9, : (due punti), _ (trattino basso) e - (trattino).

- (Facoltativo) `unit` — L'unità scientifica del segnale, ad esempio km o gradi centigradi.
- (Facoltativo) `min`: il valore minimo del segnale.
- (Facoltativo) `max`: il valore massimo del segnale.
- (Facoltativo) `assignedValue` — Il valore assegnato al segnale.
- (Facoltativo) `allowedValues`: elenco di valori accettati dal segnale.
- (Facoltativo) `deprecationMessage` — Il messaggio di deprecazione per il nodo o il ramo che viene spostato o eliminato.

Il `DeprecationMessage` può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9, : (due punti), _ (trattino basso) e - (trattino basso).

- (Facoltativo) `comment` — Un commento in aggiunta alla descrizione. Un commento può essere utilizzato per fornire informazioni aggiuntive sul sensore o sull'attuatore, ad esempio la motivazione o i riferimenti a sensori o attuatori correlati.

Il commento può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9, : (due punti), _ (trattino basso) e - (trattino).

Configura tipi di dati complessi

I tipi di dati complessi vengono utilizzati per la modellazione dei sistemi di visione. Oltre ai rami, questi tipi di dati sono costituiti da strutture (note anche come struttura) e proprietà. Una struttura è un segnale descritto da più valori, come un'immagine. Una proprietà rappresenta un membro della struttura, come un tipo di dati primitivo (come UINT8) o un'altra struttura (come timestamp). Ad esempio, `Vehicle.Cameras.Front` rappresenta un ramo, `Vehicle.Cameras.Front.Image` rappresenta una struttura e `Vehicle.Cameras.Timestamp` rappresenta una proprietà.

Il seguente esempio di tipo di dati complesso mostra come i segnali e i tipi di dati vengono esportati in un singolo file JSON.

Example tipo di dati complesso

```
{
  "Vehicle": {
    "type": "branch"
    // Signal tree
  },
  "ComplexDataTypes": {
    "VehicleDataTypes": {
      // complex data type tree
      "children": {
        "branch": {
          "children": {
            "Struct": {
              "children": {
                "Property": {
                  "type": "property",
                  "datatype": "Data type",
                  "description": "Description",
                  //          ...
                }
              },
              "description": "Description",
              "type": "struct"
            }
          }
        }
      }
    }
  }
}
```

```

        "description": "Description",
        "type": "branch"
    }
}
}
}
}

```

Note

È possibile scaricare uno [script dimostrativo](#) per convertire i messaggi ROS 2 in file VSS JSON compatibili con il catalogo dei segnali. Per ulteriori informazioni, consulta la [Vision System Data Developer Guide](#).

I dati del sistema Vision sono in versione di anteprima e sono soggetti a modifiche.

Configura la struttura

Per configurare una struttura (o struttura) personalizzata, specificare le seguenti informazioni.

- `fullyQualifiedName`— Il nome completo della struttura personalizzata.

Ad esempio, il nome completo di una struttura personalizzata potrebbe essere `ComplexDataTypes.VehicleDataTypes.SVMCamera`.

Il nome completo può contenere fino a 150 caratteri. Caratteri validi: `a—z`, `A—Z`, `0—9`, `:` (due punti) e `_` (trattino basso).

- (Facoltativo) `Description` — La descrizione del segnale.

La descrizione può contenere fino a 2048 caratteri. Caratteri validi: `a—z`, `A—Z`, `0—9`, `:` (due punti), `_` (trattino basso) e `-` (trattino).

- (Facoltativo) `deprecationMessage` — Il messaggio di deprecazione per il nodo o il ramo che viene spostato o eliminato.

Il `DeprecationMessage` può contenere fino a 2048 caratteri. Caratteri validi: `a—z`, `A—Z`, `0—9`, `:` (due punti), `_` (trattino basso) e `-` (trattino basso).

- (Facoltativo) `comment` — Un commento in aggiunta alla descrizione. Un commento può essere utilizzato per fornire informazioni aggiuntive sul sensore o sull'attuatore, ad esempio la motivazione o i riferimenti a sensori o attuatori correlati.

Il commento può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9,: (due punti), _ (trattino basso) e - (trattino).

Configura la proprietà

Per configurare una proprietà personalizzata, specificare le seguenti informazioni.

- `dataType`— Il tipo di dati del segnale deve essere uno dei seguenti: INT8, UINT8, INT16, UINT32, INT64, UINTE64, BOOLEAN, FLOAT, DOUBLE, STRING, UNIX_TIMESTAMP, INT8_ARRAY, UINTE8_ARRAY, INT16_ARRAY, UINTE16_ARRAY, UINTE16_ARRAY, INT32_ARRAY, INT_64ARRAY, INT_64_ARRAY ARRAY, UINTE64_ARRAY, BOOLEAN_ARRAY, FLOAT_ARRAY, DOUBLE_ARRAY, STRING_ARRAY, UNIX_TIMESTAMP_ARRAY, STRUCT, STRUCT_ARRAY o UNKNOWN.
- `fullyQualified_name`— Il nome completo della proprietà personalizzata. Ad esempio, il nome completo di una proprietà personalizzata potrebbe essere `ComplexDataTypes.VehicleDataTypes.SVMCamera.FPS`.

Il nome completo può contenere fino a 150 caratteri. Caratteri validi: a—z, A—Z, 0—9,: (due punti) e _ (trattino basso)

- (Facoltativo) `Description`: la descrizione del segnale.

La descrizione può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9,: (due punti), _ (trattino basso) e - (trattino).

- (Facoltativo) `deprecationMessage` — Il messaggio di deprecazione per il nodo o il ramo che viene spostato o eliminato.

Il `DeprecationMessage` può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9,: (due punti), _ (trattino basso) e - (trattino basso).

- (Facoltativo) `comment` — Un commento in aggiunta alla descrizione. Un commento può essere utilizzato per fornire informazioni aggiuntive sul sensore o sull'attuatore, ad esempio la motivazione o i riferimenti a sensori o attuatori correlati.

Il commento può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9,: (due punti), _ (trattino basso) e - (trattino).

- (Facoltativo) `dataEncoding` — Indica se la proprietà è costituita da dati binari. La codifica dei dati della proprietà personalizzata deve essere una delle seguenti: BINARY o TYPED.

- (Facoltativo) `structFullyQualifiedName` — Il nome completo del nodo della struttura (struct) per la proprietà personalizzata se il tipo di dati della proprietà personalizzata è Struct o StructArray

Il nome completo può contenere fino a 150 caratteri. Caratteri validi: a—z, A—Z, 0—9, : (due punti) e _ (trattino basso).

Crea un catalogo di segnali (AWS CLI)

È possibile utilizzare l'operazione [CreateSignalCatalog](#) API per creare un catalogo di segnali. L'esempio seguente utilizza AWS CLI.

Per creare un catalogo di segnali, esegui il comando seguente.

Sostituisci *signal-catalog-configuration* con il nome del file JSON che contiene la configurazione.

```
aws iotfleetwise create-signal-catalog --cli-input-json file://signal-catalog-configuration.json
```

- Sostituisci *signal-catalog-name* con il nome del catalogo dei segnali che stai creando.
- (Facoltativo) Sostituisci la *descrizione* con una descrizione per aiutarti a identificare il catalogo dei segnali.

Per ulteriori informazioni su come configurare rami, attributi, sensori e attuatori, vedere [Configura i segnali](#).

```
{
  "name": "signal-catalog-name",
  "description": "description",
  "nodes": [
    {
      "branch": {
        "fullyQualifiedName": "Types"
      }
    },
    {
      "struct": {
        "fullyQualifiedName": "Types.sensor_msgs_msg_CompressedImage"
      }
    }
  ]
}
```

```

},
{
  "struct": {
    "fullyQualifiedName": "Types.std_msgs_Header"
  }
},
{
  "struct": {
    "fullyQualifiedName": "Types.builtin_interfaces_Time"
  }
},
{
  "property": {
    "fullyQualifiedName": "Types.builtin_interfaces_Time.sec",
    "dataType": "INT32",
    "dataEncoding": "TYPED"
  }
},
{
  "property": {
    "fullyQualifiedName": "Types.builtin_interfaces_Time.nanosec",
    "dataType": "UINT32",
    "dataEncoding": "TYPED"
  }
},
{
  "property": {
    "fullyQualifiedName": "Types.std_msgs_Header.stamp",
    "dataType": "STRUCT",
    "structFullyQualifiedName": "Types.builtin_interfaces_Time"
  }
},
{
  "property": {
    "fullyQualifiedName": "Types.std_msgs_Header.frame_id",
    "dataType": "STRING",
    "dataEncoding": "TYPED"
  }
},
{
  "property": {
    "fullyQualifiedName": "Types.sensor_msgs_msg_CompressedImage.header",
    "dataType": "STRUCT",
    "structFullyQualifiedName": "Types.std_msgs_Header"
  }
}

```

```
    }
  },
  {
    "property": {
      "fullyQualifiedName": "Types.sensor_msgs_msg_CompressedImage.format",
      "dataType": "STRING",
      "dataEncoding": "TYPED"
    }
  },
  {
    "property": {
      "fullyQualifiedName": "Types.sensor_msgs_msg_CompressedImage.data",
      "dataType": "UINT8_ARRAY",
      "dataEncoding": "BINARY"
    }
  },
  {
    "branch": {
      "fullyQualifiedName": "Vehicle",
      "description": "Vehicle"
    }
  },
  {
    "branch": {
      "fullyQualifiedName": "Vehicle.Cameras"
    }
  },
  {
    "branch": {
      "fullyQualifiedName": "Vehicle.Cameras.Front"
    }
  },
  {
    "sensor": {
      "fullyQualifiedName": "Vehicle.Cameras.Front.Image",
      "dataType": "STRUCT",
      "structFullyQualifiedName": "Types.sensor_msgs_msg_CompressedImage"
    }
  },
  {
    "struct": {
      "fullyQualifiedName": "Types.std_msgs_msg_Float64"
    }
  },
},
```

```
{
  "property": {
    "fullyQualifiedName": "Types.std_msgs_msg_Float64.data",
    "dataType": "DOUBLE",
    "dataEncoding": "TYPED"
  }
},
{
  "sensor": {
    "fullyQualifiedName": "Vehicle.Velocity",
    "dataType": "STRUCT",
    "structFullyQualifiedName": "Types.std_msgs_msg_Float64"
  }
},
{
  "struct": {
    "fullyQualifiedName": "Types.sensor_msgs_msg_RegionOfInterest"
  }
},
{
  "property": {
    "fullyQualifiedName": "Types.sensor_msgs_msg_RegionOfInterest.x_offset",
    "dataType": "UINT32",
    "dataEncoding": "TYPED"
  }
},
{
  "property": {
    "fullyQualifiedName": "Types.sensor_msgs_msg_RegionOfInterest.y_offset",
    "dataType": "UINT32",
    "dataEncoding": "TYPED"
  }
},
{
  "property": {
    "fullyQualifiedName": "Types.sensor_msgs_msg_RegionOfInterest.height",
    "dataType": "UINT32",
    "dataEncoding": "TYPED"
  }
},
{
  "property": {
    "fullyQualifiedName": "Types.sensor_msgs_msg_RegionOfInterest.width",
    "dataType": "UINT32",

```

```
    "dataEncoding": "TYPED"
  }
},
{
  "property": {
    "fullyQualifiedName": "Types.sensor_msgs_msg_RegionOfInterest.do_rectify",
    "dataType": "BOOLEAN",
    "dataEncoding": "TYPED"
  }
},
{
  "branch": {
    "fullyQualifiedName": "Vehicle.Perception"
  }
},
{
  "sensor": {
    "fullyQualifiedName": "Vehicle.Perception.Obstacle",
    "dataType": "STRUCT",
    "structFullyQualifiedName": "Types.sensor_msgs_msg_RegionOfInterest"
  }
}
]
}
```

Note

È possibile scaricare uno [script dimostrativo](#) per convertire i messaggi ROS 2 in file VSS JSON compatibili con il catalogo dei segnali. Per ulteriori informazioni, consulta la [Vision System Data Developer Guide](#).

I dati del sistema Vision sono in versione di anteprima e sono soggetti a modifiche.

Importa un catalogo di segnali

Puoi utilizzare la FleetWise console o l'API AWS IoT per importare un catalogo di segnali.

Argomenti

- [Importa un catalogo di segnali \(console\)](#)
- [Importa un catalogo di segnali \(AWS CLI\)](#)

Importa un catalogo di segnali (console)

Puoi utilizzare la FleetWise console AWS IoT per importare un catalogo di segnali.

Important

È possibile avere al massimo un catalogo di segnali. Se disponi già di un catalogo di segnali, non vedrai l'opzione per importare un catalogo di segnali nella console.

Per importare un catalogo di segnali

1. Apri la [FleetWise console AWS IoT](#).
2. Nel pannello di navigazione, scegli Signal catalog.
3. Nella pagina di riepilogo del catalogo dei segnali, scegli Importa catalogo dei segnali.
4. Importa il file contenente i segnali.
 - Per caricare un file da un bucket S3:
 - a. Scegliere Import from S3 (Importazione da S3).
 - b. Seleziona Sfoglia S3.
 - c. Per Bucket, inserisci il nome o l'oggetto del bucket, scegliilo dall'elenco, quindi scegli il file dall'elenco. Scegli il pulsante Scegli file.

Oppure, per l'URI S3, inserisci un URI di Amazon Simple Storage Service. Per ulteriori informazioni, consulta [Metodi per l'accesso a un bucket](#) nella Amazon S3 User Guide.
 - Per caricare un file dal tuo computer:
 - a. Scegli Importa da file.
 - b. Carica un file.json in un formato [VSS \(Vehicle Signal Specification\)](#).
5. Verifica il catalogo dei segnali, quindi scegli Importa file.

Importa un catalogo di segnali (AWS CLI)

Puoi utilizzare l'operazione [ImportSignalCatalog](#) API per caricare un file JSON che aiuta a creare un catalogo di segnali. È necessario seguire la [Vehicle Signal Specification \(VSS\)](#) per salvare i segnali nel file JSON. L'esempio seguente utilizza. AWS CLI

Per importare un catalogo di segnali, esegui il comando seguente.

- Sostituiscilo *signal-catalog-name* con il nome del catalogo di segnali che stai creando.
- (Facoltativo) Sostituisci la *descrizione con una descrizione* per aiutarti a identificare il catalogo dei segnali.
- Sostituisci *signal-catalog-configuration-vss* con il nome del file di stringa JSON che contiene i segnali definiti in VSS.

Per ulteriori informazioni su come configurare rami, attributi, sensori e attuatori, vedere. [Configura i segnali](#)

```
aws iotfleetwise import-signal-catalog \  
    --name signal-catalog-name \  
    --description description \  
    --vss file://signal-catalog-configuration-vss.json
```

Il codice JSON deve essere stringato e passato attraverso il campo. `vssJson` Di seguito è riportato un esempio di segnali definiti in VSS.

```
{  
  "Vehicle": {  
    "type": "branch",  
    "children": {  
      "Chassis": {  
        "type": "branch",  
        "description": "All data concerning steering, suspension, wheels, and brakes.",  
        "children": {  
          "SteeringWheel": {  
            "type": "branch",  
            "description": "Steering wheel signals",  
            "children": {  
              "Diameter": {  
                "type": "attribute",  
                "description": "The diameter of the steering wheel",  
                "datatype": "float",  
                "unit": "cm",  
                "min": 1,  
                "max": 50  
              },  
              "HandsOff": {
```

```

    "type": "branch",
    "children": {
      "HandsOffSteeringState": {
        "type": "actuator",
        "description": "HndsOffStrWhlDtSt. Hands Off Steering State",
        "datatype": "boolean"
      },
      "HandsOffSteeringMode": {
        "type": "actuator",
        "description": "HndsOffStrWhlDtMd. Hands Off Steering Mode",
        "datatype": "int8",
        "min": 0,
        "max": 2
      }
    }
  },
  "Accelerator": {
    "type": "branch",
    "description": "",
    "children": {
      "AcceleratorPedalPosition": {
        "type": "sensor",
        "description": "Throttle__Position. Accelerator pedal position as percent. 0 =
Not depressed. 100 = Fully depressed.",
        "datatype": "uint8",
        "unit": "%",
        "min": 0,
        "max": 100.000035
      }
    }
  },
  "Powertrain": {
    "type": "branch",
    "description": "Powertrain data for battery management, etc.",
    "children": {
      "Transmission": {
        "type": "branch",
        "description": "Transmission-specific data, stopping at the drive shafts.",
        "children": {
          "VehicleOdometer": {

```

```
    "type": "sensor",
    "description": "Vehicle_Odometer",
    "datatype": "float",
    "unit": "km",
    "min": 0,
    "max": 67108863.984375
  }
}
},
"CombustionEngine": {
  "type": "branch",
  "description": "Engine-specific data, stopping at the bell housing.",
  "children": {
    "Engine": {
      "type": "branch",
      "description": "Engine description",
      "children": {
        "timing": {
          "type": "branch",
          "description": "timing description",
          "children": {
            "run_time": {
              "type": "sensor",
              "description": "Engine run time",
              "datatype": "int16",
              "unit": "ms",
              "min": 0,
              "max": 10000
            },
            "idle_time": {
              "type": "sensor",
              "description": "Engine idle time",
              "datatype": "int16",
              "min": 0,
              "unit": "ms",
              "max": 10000
            }
          }
        }
      }
    }
  }
}
```

```
},
"Axle": {
  "type": "branch",
  "description": "Axle signals",
  "children": {
    "TireRRPrs": {
      "type": "sensor",
      "description": "TireRRPrs. Right rear Tire pressure in kilo-Pascal",
      "datatype": "float",
      "unit": "kPaG",
      "min": 0,
      "max": 1020
    }
  }
}
},
"Cameras": {
  "type": "branch",
  "description": "Branch to aggregate all cameras in the vehicle",
  "children": {
    "FrontViewCamera": {
      "type": "sensor",
      "datatype": "VehicleDataTypes.SVMCamera",
      "description": "Front view camera"
    },
    "RearViewCamera": {
      "type": "sensor",
      "datatype": "VehicleDataTypes.SVMCamera",
      "description": "Rear view camera"
    },
    "LeftSideViewCamera": {
      "type": "sensor",
      "datatype": "VehicleDataTypes.SVMCamera",
      "description": "Left side view camera"
    },
    "RightSideViewCamera": {
      "type": "sensor",
      "datatype": "VehicleDataTypes.SVMCamera",
      "description": "Right side view camera"
    }
  }
}
},
"ComplexDataTypes": {
```

```
"VehicleDataTypes": {
  "type": "branch",
  "description": "Branch to aggregate all camera related higher order data types",
  "children": {
    "SVMCamera": {
      "type": "struct",
      "description": "This data type represents Surround View Monitor (SVM) camera
system in a vehicle",
      "comment": "Test comment",
      "deprecation": "Test deprecation message",
      "children": {
        "Make": {
          "type": "property",
          "description": "Make of the SVM camera",
          "datatype": "string",
          "comment": "Test comment",
          "deprecation": "Test deprecation message"
        },
        "Description": {
          "type": "property",
          "description": "Description of the SVM camera",
          "datatype": "string",
          "comment": "Test comment",
          "deprecation": "Test deprecation message"
        },
        "FPS": {
          "type": "property",
          "description": "FPS of the SVM camera",
          "datatype": "double",
          "comment": "Test comment",
          "deprecation": "Test deprecation message"
        },
        "Orientation": {
          "type": "property",
          "description": "Orientation of the SVM camera",
          "datatype": "VehicleDataTypes.Orientation",
          "comment": "Test comment",
          "deprecation": "Test deprecation message"
        },
        "Range": {
          "type": "property",
          "description": "Range of the SVM camera",
          "datatype": "VehicleDataTypes.Range",
          "comment": "Test comment",
```

```
    "deprecation": "Test deprecation message"
  },
  "RawData": {
    "type": "property",
    "description": "Represents binary data of the SVM camera",
    "datatype": "uint8[]",
    "dataencoding": "binary",
    "comment": "Test comment",
    "deprecation": "Test deprecation message"
  },
  "CapturedFrames": {
    "type": "property",
    "description": "Represents selected frames captured by the SVM camera",
    "datatype": "VehicleDataTypes.Frame[]",
    "dataencoding": "typed",
    "comment": "Test comment",
    "deprecation": "Test deprecation message"
  }
}
},
"Range": {
  "type": "struct",
  "description": "Range of a camera in centimeters",
  "comment": "Test comment",
  "deprecation": "Test deprecation message",
  "children": {
    "Min": {
      "type": "property",
      "description": "Minimum range of a camera in centimeters",
      "datatype": "uint32",
      "comment": "Test comment",
      "deprecation": "Test deprecation message"
    },
    "Max": {
      "type": "property",
      "description": "Maximum range of a camera in centimeters",
      "datatype": "uint32",
      "comment": "Test comment",
      "deprecation": "Test deprecation message"
    }
  }
},
"Orientation": {
  "type": "struct",
```

```
"description": "Orientation of a camera",
"comment": "Test comment",
"deprecation": "Test deprecation message",
"children": {
  "Front": {
    "type": "property",
    "description": "Indicates whether the camera is oriented to the front of the
vehicle",
    "datatype": "boolean",
    "comment": "Test comment",
    "deprecation": "Test deprecation message"
  },
  "Rear": {
    "type": "property",
    "description": "Indicates whether the camera is oriented to the rear of the
vehicle",
    "datatype": "boolean",
    "comment": "Test comment",
    "deprecation": "Test deprecation message"
  },
  "Side": {
    "type": "property",
    "description": "Indicates whether the camera is oriented to the side of the
vehicle",
    "datatype": "boolean",
    "comment": "Test comment",
    "deprecation": "Test deprecation message"
  }
},
"Frame": {
  "type": "struct",
  "description": "Represents a camera frame",
  "comment": "Test comment",
  "deprecation": "Test deprecation message",
  "children": {
    "Data": {
      "type": "property",
      "datatype": "string",
      "dataencoding": "binary",
      "comment": "Test comment",
      "deprecation": "Test deprecation message"
    }
  }
}
```

```

    }
  }
}
}
}

```

L'esempio seguente mostra gli stessi segnali definiti in VSS in una stringa JSON.

```

{
  "vssJson": "{ \"Vehicle\": { \"type\": \"branch\", \"children\": { \"Chassis\": { \"type\": \"branch\", \"description\": \"All data concerning steering, suspension, wheels, and brakes.\", \"children\": { \"SteeringWheel\": { \"type\": \"branch\", \"description\": \"Steering wheel signals\", \"children\": { \"Diameter\": { \"type\": \"attribute\", \"description\": \"The diameter of the steering wheel\", \"datatype\": \"float\", \"unit\": \"cm\", \"min\": 1, \"max\": 50}, \"HandsOff\": { \"type\": \"branch\", \"children\": { \"HandsOffSteeringState\": { \"type\": \"actuator\", \"description\": \"HndsOffStrWhlDtSt. Hands Off Steering State\", \"datatype\": \"boolean\"}, \"HandsOffSteeringMode\": { \"type\": \"actuator\", \"description\": \"HndsOffStrWhlDtMd. Hands Off Steering Mode\", \"datatype\": \"int8\", \"min\": 0, \"max\": 2}}}}, \"Accelerator\": { \"type\": \"branch\", \"description\": \"\", \"children\": { \"AcceleratorPedalPosition\": { \"type\": \"sensor\", \"description\": \"Throttle__Position. Accelerator pedal position as percent. 0 = Not depressed. 100 = Fully depressed.\", \"datatype\": \"uint8\", \"unit\": \"%\", \"min\": 0, \"max\": 100.000035}}}}, \"Powertrain\": { \"type\": \"branch\", \"description\": \"Powertrain data for battery management, etc.\", \"children\": { \"Transmission\": { \"type\": \"branch\", \"description\": \"Transmission-specific data, stopping at the drive shafts.\", \"children\": { \"VehicleOdometer\": { \"type\": \"sensor\", \"description\": \"Vehicle_Odometer\", \"datatype\": \"float\", \"unit\": \"km\", \"min\": 0, \"max\": 67108863.984375}}}, \"CombustionEngine\": { \"type\": \"branch\", \"description\": \"Engine-specific data, stopping at the bell housing.\", \"children\": { \"Engine\": { \"type\": \"branch\", \"description\": \"Engine description\", \"children\": { \"timing\": { \"type\": \"branch\", \"description\": \"timing description\", \"children\": { \"run_time\": { \"type\": \"sensor\", \"description\": \"Engine run time\", \"datatype\": \"int16\", \"unit\": \"ms\", \"min\": 0, \"max\": 10000}, \"idle_time\": { \"type\": \"sensor\", \"description\": \"Engine idle time\", \"datatype\": \"int16\", \"min\": 0, \"unit\": \"ms\", \"max\": 10000}}}}}}, \"Axle\": { \"type\": \"branch\", \"description\": \"Axle signals\", \"children\": { \"TireRRPrs\": { \"type\": \"sensor\", \"description\": \"TireRRPrs. Right rear Tire pressure in kilo-Pascal\", \"datatype\": \"float\", \"unit\": \"kPaG\", \"min\": 0, \"max\": 1020}}}}}}}"
}

```

Note

È possibile scaricare uno [script dimostrativo](#) per convertire i messaggi ROS 2 in file VSS JSON compatibili con il catalogo dei segnali. Per ulteriori informazioni, consulta la [Vision System Data Developer Guide](#).

I dati del sistema Vision sono in versione di anteprima e sono soggetti a modifiche.

Aggiornare un catalogo di segnali (AWS CLI)

È possibile utilizzare l'operazione [UpdateSignalCatalog](#) API per aggiornare un catalogo di segnali esistente. L'esempio seguente utilizza AWS CLI.

Per aggiornare un catalogo di segnali esistente, esegui il comando seguente.

Sostituisci *signal-catalog-configuration* con il nome del file JSON che contiene la configurazione.

```
aws iotfleetwise update-signal-catalog --cli-input-json file://signal-catalog-configuration.json
```

Sostituisci *signal-catalog-name* con il nome del catalogo dei segnali che stai aggiornando.

Per ulteriori informazioni su come configurare rami, attributi, sensori e attuatori, consulta [Configura i segnali](#).

Important

Le strutture personalizzate sono immutabili. Se devi riordinare o inserire proprietà in una struttura personalizzata esistente (struct), elimina la struttura e crea una nuova struttura con l'ordine di proprietà desiderato.

Per eliminare una struttura personalizzata, aggiungi il nome completo della struttura in `nodesToRemove`. Una struttura non può essere eliminata se ad essa fanno riferimento dei segnali. Tutti i segnali che fanno riferimento alla struttura (il loro tipo di dati è definito come struttura di destinazione) devono essere aggiornati o eliminati prima della richiesta di aggiornamento del catalogo dei segnali.

```
{
  "name": "signal-catalog-name",
  "nodesToAdd": [{
    "branch": {
      "description": "Front left of vehicle specific data.",
      "fullyQualifiedName": "Vehicle.Front.Left"
    }
  },
  {
    "branch": {
      "description": "Door-specific data for the front left of vehicle.",
      "fullyQualifiedName": "Vehicle.Front.Left.Door"
    }
  },
  {
    "actuator": {
      "fullyQualifiedName": "Vehicle.Front.Left.Door.Lock",
      "description": "Whether the front left door is locked.",
      "dataType": "BOOLEAN"
    }
  },
  {
    "branch": {
      "fullyQualifiedName": "Vehicle.Camera"
    }
  },
  {
    "struct": {
      "fullyQualifiedName": "Vehicle.Camera.SVMCamera"
    }
  },
  {
    "property": {
      "fullyQualifiedName": "Vehicle.Camera.SVMCamera.ISO",
      "dataType": "STRING"
    }
  }
  ],
  "nodesToRemove": ["Vehicle.Chassis.SteeringWheel.HandsOffSteeringState"],
  "nodesToUpdate": [{
    "attribute": {
      "dataType": "FLOAT",
      "fullyQualifiedName": "Vehicle.Chassis.SteeringWheel.Diameter",
```

```
    "max": 55
  }
}]
}
```

Eliminare un catalogo di segnali (AWS CLI)

È possibile utilizzare l'operazione [DeleteSignalCatalog](#) API per eliminare un catalogo di segnali. L'esempio seguente utilizza AWS CLI.

Important

Prima di eliminare un catalogo di segnali, assicuratevi che non contenga modelli di veicoli, manifesti di decoder, veicoli, flotte o campagne associati. Per le istruzioni, consulta quanto segue:

- [Eliminare un modello di veicolo](#)
- [Eliminare un manifesto del decodificatore](#)
- [Eliminare un veicolo](#)
- [Elimina un parco istanze \(AWS CLI\)](#)
- [Eliminare una campagna](#)

Per eliminare un catalogo di segnali esistente, esegui il comando seguente. Sostituiscilo *signal-catalog-name* con il nome del catalogo dei segnali che stai eliminando.

```
aws iotfleetwise delete-signal-catalog --name signal-catalog-name
```

Note

Questo comando non produce output.

Ottieni informazioni sul catalogo dei segnali (AWS CLI)

È possibile utilizzare l'operazione [ListSignalCatalogs](#) API per verificare se un catalogo di segnali è stato eliminato. L'esempio seguente utilizza AWS CLI.

Per recuperare un elenco impaginato di riepiloghi di tutti i cataloghi di segnali, esegui il comando seguente.

```
aws iotfleetwise list-signal-catalogs
```

È possibile utilizzare l'operazione [ListSignalCatalogNodes](#) API per verificare se un catalogo di segnali è stato aggiornato. L'esempio seguente utilizza AWS CLI.

Per recuperare un elenco impaginato di riepiloghi di tutti i segnali (nodi) in un determinato catalogo di segnali, esegui il comando seguente.

signal-catalog-name Sostituiscilo con il nome del catalogo dei segnali che stai controllando.

```
aws iotfleetwise list-signal-catalog-nodes --name signal-catalog-name
```

Puoi utilizzare l'operazione [GetSignalCatalog](#) API per recuperare le informazioni del catalogo dei segnali. L'esempio seguente utilizza AWS CLI.

Per recuperare informazioni su un catalogo di segnali, esegui il comando seguente.

Sostituirelo *signal-catalog-name* con il nome del catalogo dei segnali che desiderate recuperare.

```
aws iotfleetwise get-signal-catalog --name signal-catalog-name
```

Note

Questa operazione è [consistente finale](#). In altre parole, le modifiche al catalogo dei segnali potrebbero non essere applicate immediatamente.

Crea e gestisci modelli di veicoli

Utilizzi i segnali per creare modelli di veicoli che aiutano a standardizzare il formato dei veicoli. I modelli di veicoli applicano informazioni coerenti su più veicoli dello stesso tipo, in modo da poter elaborare i dati provenienti da flotte di veicoli. I veicoli creati con lo stesso modello di veicolo ereditano lo stesso gruppo di segnali. Per ulteriori informazioni, consulta [Crea, fornisci e gestisci veicoli](#).

Ogni modello di veicolo ha un campo di stato che contiene lo stato del modello di veicolo. Lo stato può avere uno dei seguenti valori:

- ACTIVE— Il modello di veicolo è attivo.
- DRAFT— La configurazione del modello di veicolo viene salvata.

Important

- Se desideri utilizzare l'operazione `CreateModelManifest` API per creare il primo modello di veicolo, devi prima creare un catalogo di segnali. Per ulteriori informazioni, consulta [Crea un catalogo di segnali \(\)AWS CLI](#).
- Se utilizzi la FleetWise console AWS IoT per creare un modello di veicolo, AWS IoT attiva FleetWise automaticamente il modello di veicolo per te.
- Se utilizzi l'operazione `CreateModelManifest` API per creare un modello di veicolo, il modello di veicolo rimane invariato. DRAFT
- Non puoi creare veicoli a partire da modelli di veicoli che si trovano nello DRAFT stato. Utilizza l'operazione `UpdateModelManifest` API per modificare i modelli di veicoli in base allo ACTIVE stato.
- Non puoi modificare i modelli di veicoli che si trovano nello ACTIVE stato.

Argomenti

- [Crea un modello di veicolo](#)
- [Aggiornare un modello di veicolo \(\)AWS CLI](#)
- [Eliminare un modello di veicolo](#)
- [Ottieni informazioni sul modello del veicolo \(AWS CLI\)](#)

Crea un modello di veicolo

Puoi utilizzare la FleetWise console o l'API AWS IoT per creare modelli di veicoli.

Important

È necessario disporre di un catalogo di segnali prima di poter creare un modello di veicolo utilizzando l'operazione `CreateModelManifest` API.

Argomenti

- [Crea un modello di veicolo \(console\)](#)
- [Crea un modello di veicolo \(AWS CLI\)](#)

Crea un modello di veicolo (console)

Nella FleetWise console AWS IoT, puoi creare un modello di veicolo nei seguenti modi:

- [Utilizza un modello fornito da AWS](#)
- [Crea manualmente un modello di veicolo](#)
- [Duplica un modello di veicolo](#)

Utilizza un modello fornito da AWS

AWS FleetWise L'iot fornisce un modello di diagnostica di bordo (OBD) II, J1979 che crea automaticamente un catalogo di segnali, un modello di veicolo e un manifesto di decodifica. Il modello aggiunge anche interfacce di rete OBD al manifesto del decoder. Per ulteriori informazioni, consulta [Crea e gestisci i manifesti del decodificatore](#).

Per creare un modello di veicolo utilizzando un modello

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. Nel pannello di navigazione, scegli Modelli di veicoli.
3. Nella pagina Modelli di veicoli, scegli Aggiungi modello fornito.
4. Scegli Diagnostica di bordo (OBD) II.
5. Immettere un nome per l'interfaccia di rete OBD creata dall' AWS IoT FleetWise .
6. Scegli Aggiungi.

Crea manualmente un modello di veicolo

Puoi aggiungere segnali dal catalogo dei segnali o importare segnali caricando uno o più file.dbc. Un file.dbc è un formato di file supportato dai database Controller Area Network (CAN bus).

⚠ Important

Non è possibile creare un modello di veicolo con segnali di dati del sistema di visione utilizzando la FleetWise console AWS IoT. Utilizza invece il AWS CLI per creare un modello di veicolo.

I dati del sistema di visione sono disponibili in anteprima e sono soggetti a modifiche.

Per creare manualmente un modello di veicolo

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. Nel pannello di navigazione, scegli Modelli di veicoli.
3. Nella pagina Modelli di veicolo, scegli Crea modello di veicolo, quindi procedi come segue.

Argomenti

- [Passaggio 1: configura il modello del veicolo](#)
- [Fase 2: Aggiungere segnali](#)
- [Fase 3: Importazione dei segnali](#)
- [\(Facoltativo\) Fase 4: Aggiungere attributi](#)
- [Passaggio 5: revisione e creazione](#)

Passaggio 1: configura il modello del veicolo

In Informazioni generali, procedi come segue.

1. Inserisci un nome per il modello del veicolo.
2. (Opzionale) Immettere una descrizione.
3. Seleziona Successivo.

Fase 2: Aggiungere segnali

i Note

- Se è la prima volta che utilizzi l' AWS IoT FleetWise, questo passaggio non è disponibile finché non disponi di un catalogo di segnali. Quando viene creato il primo modello di

veicolo, l' AWS IoT crea FleetWise automaticamente un catalogo di segnali con segnali aggiunti al primo modello di veicolo.

- Se hai esperienza con l' AWS IoT FleetWise, puoi aggiungere segnali al tuo modello di veicolo selezionando i segnali dal catalogo dei segnali o caricando file.dbc per importare segnali.
- È necessario disporre di almeno un segnale per creare un modello di veicolo.

Per aggiungere segnali

1. Scegli uno o più segnali dal catalogo dei segnali che stai aggiungendo al modello di veicolo. Puoi esaminare i segnali selezionati nel riquadro a destra.

Note

Solo i segnali selezionati verranno aggiunti al modello del veicolo.

2. Seleziona Successivo.

Fase 3: Importazione dei segnali

Note

- Se è la prima volta che utilizzi l' AWS IoT FleetWise, devi caricare almeno un file.dbc per importare i segnali.
- Se hai esperienza con l' AWS IoT FleetWise, puoi aggiungere segnali al tuo modello di veicolo selezionando i segnali dal catalogo dei segnali o caricando file.dbc per importare segnali.
- È necessario disporre di almeno un segnale per creare un modello di veicolo.

Per importare segnali

1. Scegli Scegli file.
2. Nella finestra di dialogo, scegliete il file.dbc che contiene i segnali. È possibile caricare più file.dbc.

3. AWS L'IoT FleetWise analizza i tuoi file.dbc per recuperare i segnali.

Nella sezione Segnali, specifica i seguenti metadati per ogni segnale.

- Nome: il nome del segnale.

Il nome del segnale deve essere univoco. Il nome del segnale e il percorso possono contenere fino a 150 caratteri. Caratteri validi: a—z, A—Z, 0—9,: (due punti) e _ (trattino basso).

- Tipo di dati: il tipo di dati del segnale deve essere uno dei seguenti: INT8, UINT8, INT16, INT32, UINT32, INT64, UINT64, BOOLEAN, FLOAT, DOUBLE, STRING, UNIX_TIMESTAMP, INT8_ARRAY, UINT8_ARRAY, INT16_ARRAY, UINT16_ARRAY, INT32_ARRAY, INT32_ARRAY, INT32_ARRAY, 64_ARRAY, UINT64_ARRAY, BOOLEAN_ARRAY, FLOAT_ARRAY, DOUBLE_ARRAY, STRING_ARRAY, UNIX_TIMESTAMP_ARRAY o UNKNOWN.
- Tipo di segnale: il tipo di segnale, che può essere sensore o attuatore.
- (Facoltativo) Unità: l'unità scientifica per il segnale, ad esempio km o Celsius.
- (Facoltativo) Percorso: il percorso verso il segnale. Analogamente a JSONPath, utilizzate un punto (.) per fare riferimento a un segnale secondario. Ad esempio, **Vehicle.Engine.Light**.

Il nome del segnale più il percorso possono contenere fino a 150 caratteri. Caratteri validi: a—z, A—Z, 0—9,: (due punti) e _ (trattino basso).

- (Facoltativo) Min: il valore minimo del segnale.
- (Facoltativo) Max: il valore massimo del segnale.
- (Facoltativo) Descrizione: la descrizione del segnale.

La descrizione può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9,: (due punti), _ (trattino basso) e - (trattino).

4. Seleziona Successivo.

(Facoltativo) Fase 4: Aggiungere attributi

È possibile aggiungere fino a 100 attributi, inclusi gli attributi esistenti nel catalogo dei segnali.

Per aggiungere attributi

1. In Aggiungi attributi, specificate i seguenti metadati per ogni attributo.

- **Nome:** il nome dell'attributo.

Il nome del segnale deve essere univoco. Il nome e il percorso del segnale possono contenere fino a 150 caratteri. Caratteri validi: a—z, A—Z, 0—9,: (due punti) e _ (trattino basso)

- **Tipo di dati:** il tipo di dati dell'attributo deve essere uno dei seguenti: INT8, UINT8, INT16, INT32, UINT32, INT64, UINT64, BOOLEAN, FLOAT, DOUBLE, STRING, UNIX_TIMESTAMP, INT8_ARRAY, UINT8_ARRAY, INT16_ARRAY, UINT16_ARRAY, INT32_ARRAY, INT32_ARRAY, 64_ARRAY, UINT64_ARRAY, BOOLEAN_ARRAY, FLOAT_ARRAY, DOUBLE_ARRAY, STRING_ARRAY, UNIX_TIMESTAMP_ARRAY o UNKNOWN
- (Facoltativo) **Unità:** l'unità scientifica per l'attributo, ad esempio km o Celsius.
- (Facoltativo) **Percorso:** il percorso verso il segnale. Analogamente a JSONPath, utilizzate un punto (.) per fare riferimento a un segnale secondario. Ad esempio, **Vehicle.Engine.Light**.

Il nome del segnale più il percorso possono contenere fino a 150 caratteri. Caratteri validi: a—z, A—Z, 0—9,: (due punti) e _ (trattino basso)

- (Facoltativo) **Min:** il valore minimo dell'attributo.
- (Facoltativo) **Max:** il valore massimo dell'attributo.
- (Facoltativo) **Descrizione:** la descrizione dell'attributo.

La descrizione può contenere fino a 2048 caratteri. Caratteri validi: a—z, A—Z, 0—9,: (due punti), _ (trattino basso) e - (trattino).

2. Seleziona Successivo.

Passaggio 5: revisione e creazione

Verifica le configurazioni per il modello di veicolo, quindi scegli Crea.

Duplica un modello di veicolo

AWS L'IoT FleetWise può copiare le configurazioni di un modello di veicolo esistente per crearne uno nuovo. I segnali specificati nel modello di veicolo selezionato vengono copiati nel nuovo modello di veicolo.

Per duplicare un modello di veicolo

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. Nel pannello di navigazione, scegli Modelli di veicoli.
3. Scegli un modello dall'elenco dei modelli di veicolo, quindi scegli Modello duplicato.

Per configurare il modello del veicolo, segui il [Crea manualmente un modello di veicolo](#) tutorial.

L'elaborazione della richiesta di creazione del modello di veicolo FleetWise da parte dell' AWS IoT può richiedere alcuni minuti. Dopo che il modello di veicolo è stato creato con successo, nella pagina Modelli di veicolo, la colonna Stato mostra ATTIVO. Quando il modello di veicolo diventa attivo, non puoi modificarlo.

Crea un modello di veicolo (AWS CLI)

È possibile utilizzare l'operazione [CreateModelManifest](#)API per creare modelli di veicoli (model manifest). Gli esempi seguenti utilizzano AWS CLI.

Important

Se desideri utilizzare l' FleetWise API AWS IoT per creare il primo modello di veicolo, devi prima creare un catalogo di segnali. Per ulteriori informazioni su come creare un catalogo di segnali, consulta [Crea un catalogo di segnali \(\)AWS CLI](#).

Per creare un modello di veicolo, esegui il comando seguente.

Sostituisci *vehicle-model-configuration* con il nome del file JSON che contiene la configurazione.

```
aws iotfleetwise create-model-manifest --cli-input-json file://vehicle-model-configuration.json
```

- Sostituisci la *vehicle-model-name* con il nome del modello di veicolo che stai creando.
- Sostituisci *Signal-catalog-ARN* con l'Amazon Resource Name (ARN) del catalogo dei segnali.
- (Facoltativo) Sostituisci la *descrizione* con una descrizione per aiutarti a identificare il modello del veicolo.

Per ulteriori informazioni su come configurare rami, attributi, sensori e attuatori, consulta [Configura i segnali](#).

```
{
  "name": "vehicle-model-name",
  "signalCatalogArn": "signal-catalog-ARN",
  "description": "description",
  "nodes": ["Vehicle.Chassis"]
}
```

Aggiornare un modello di veicolo ()AWS CLI

È possibile utilizzare l'operazione [UpdateModelManifest](#) API per aggiornare un modello di veicolo esistente (manifesti del modello). Gli esempi seguenti utilizzano AWS CLI.

Per aggiornare un modello di veicolo esistente, esegui il comando seguente.

Sostituisci *update-vehicle-model-configuration* con il nome del file JSON che contiene la configurazione.

```
aws iotfleetwise update-model-manifest --cli-input-json file://update-vehicle-model-configuration.json
```

- Sostituiscilo *vehicle-model-name* con il nome del modello di veicolo che stai aggiornando.
- (Facoltativo) Per attivare il modello di veicolo, *vehicle-model-status* sostituiscilo con ACTIVE.

Important

Dopo l'attivazione del modello di veicolo, non è possibile modificarlo.

- (Facoltativo) Sostituisci la *descrizione* con una descrizione aggiornata per aiutarti a identificare il modello del veicolo.

```
{
  "name": "vehicle-model-name",
  "status": "vehicle-model-status",
  "description": "description",
  "nodesToAdd": ["Vehicle.Front.Left"],
}
```

```
"nodesToRemove": ["Vehicle.Chassis.SteeringWheel"],  
}
```

Eliminare un modello di veicolo

Puoi utilizzare la FleetWise console o l'API AWS IoT per eliminare i modelli di veicoli.

Important

I veicoli e i manifesti del decoder associati al modello di veicolo devono essere prima eliminati. Per ulteriori informazioni, consultare [Eliminare un veicolo](#) e [Eliminare un manifesto del decodificatore](#).

Eliminare un modello di veicolo (console)

Per eliminare un modello di veicolo, utilizza la FleetWise console AWS IoT.

Per eliminare un modello di veicolo

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. Nel pannello di navigazione, scegli Modelli di veicoli.
3. Nella pagina Modelli di veicoli, scegli il modello di veicolo di destinazione.
4. Scegli Elimina.
5. In Elimina **vehicle-model-name?**, inserisci il nome del modello di veicolo da eliminare, quindi scegli Conferma.

Elimina un modello di veicolo (AWS CLI)

È possibile utilizzare l'operazione [DeleteModelManifest](#) API per eliminare un modello di veicolo esistente (manifesti del modello). Gli esempi seguenti utilizzano AWS CLI.

Per eliminare un modello di veicolo, esegui il comando seguente.

Sostituiscilo *model-manifest-name* con il nome del modello di veicolo che stai eliminando.

```
aws iotfleetwise delete-model-manifest --name model-manifest-name
```

Note

Questo comando non produce output.

Ottieni informazioni sul modello del veicolo (AWS CLI)

Puoi utilizzare l'operazione [ListModelManifests](#) API per verificare se un modello di veicolo è stato eliminato. L'esempio seguente utilizza AWS CLI.

Per recuperare un elenco impaginato di riepiloghi di tutti i modelli di veicoli, esegui il comando seguente.

```
aws iotfleetwise list-model-manifests
```

Puoi utilizzare l'operazione [ListModelManifestNodes](#) API per verificare se un modello di veicolo è stato aggiornato. L'esempio seguente utilizza AWS CLI.

Per recuperare un elenco impaginato di riepiloghi di tutti i segnali (nodi) in un determinato modello di veicolo, esegui il comando seguente.

vehicle-model-name Sostituiscilo con il nome del modello di veicolo che stai controllando.

```
aws iotfleetwise list-model-manifest-nodes /  
    --name vehicle-model-name
```

Per recuperare informazioni su un modello di veicolo, esegui il comando seguente.

Sostituisci il *modello di veicolo* con il nome del modello di veicolo che desideri recuperare.

```
aws iotfleetwise get-model-manifest --name vehicle-model
```

Note

Questa operazione è [consistente finale](#). In altre parole, le modifiche al modello del veicolo potrebbero non riflettersi immediatamente.

Crea e gestisci i manifesti del decodificatore

I manifesti di decodifica contengono informazioni di decodifica che l'AWS IoT FleetWise utilizza per trasformare i dati dei veicoli (dati binari) in valori leggibili dall'uomo e per preparare i dati per l'analisi dei dati. L'interfaccia di rete e i segnali del decodificatore sono i componenti principali con cui lavorate per configurare i manifesti del decoder.

Interfaccia di rete

Contiene informazioni sul protocollo utilizzato dalla rete di bordo. AWS L'IoT FleetWise supporta i seguenti protocolli.

Controller Area Network (bus CAN)

Un protocollo che definisce il modo in cui i dati vengono comunicati tra unità di controllo elettroniche (ECU). Le ECU possono essere l'unità di controllo del motore, gli airbag o il sistema audio.

Diagnostica di bordo (OBD) II

Un protocollo ulteriormente sviluppato che definisce il modo in cui i dati di autodiagnostica vengono comunicati tra le ECU. Fornisce una serie di codici diagnostici di errore (DTC) standard che aiutano a identificare i problemi del veicolo.

Middleware per veicoli

Il middleware del veicolo definito come un tipo di interfaccia di rete. Esempi di middleware per veicoli includono Robot Operating System (ROS 2) e Scalable Service-Oriented Middleware over IP (SOME/IP).

Note

AWS L'IoT FleetWise supporta il middleware ROS 2 per i dati dei sistemi di visione.

Segnale del decoder

Fornisce informazioni di decodifica dettagliate per un segnale specifico. Ogni segnale specificato nel modello di veicolo deve essere associato a un segnale di decodifica. Se il manifesto del decodificatore contiene interfacce di rete CAN, deve contenere segnali di decodifica CAN. Se il manifesto del decodificatore contiene interfacce di rete OBD, deve contenere segnali di decodifica OBD.

Il manifesto del decodificatore deve contenere segnali di decodifica dei messaggi se contiene anche interfacce middleware del veicolo.

Ogni manifesto del decodificatore deve essere associato a un modello di veicolo. AWS L' IoT FleetWise utilizza il manifesto del decoder associato per decodificare i dati dei veicoli creati in base al modello del veicolo.

Ogni manifesto del decodificatore ha un campo di stato che contiene lo stato del manifesto del decoder. Lo stato può avere uno dei seguenti valori:

- **ACTIVE**— Il manifesto del decodificatore è attivo.
- **DRAFT**— La configurazione del manifesto del decoder non viene salvata.
- **VALIDATING**— Il manifesto del decodificatore è in fase di convalida per verificarne l'idoneità. Questo vale solo per i manifesti del decodificatore che contengono almeno un segnale dati del sistema di visione.
- **INVALID**— Il manifesto del decoder non è riuscito a convalidare e non può ancora essere attivato. Questo vale solo per i manifesti del decoder che contengono almeno un segnale dati del sistema di visione. È possibile utilizzare le `GetDecoderManifest` API `ListDecoderManifests` and per verificare il motivo di una convalida non riuscita.

Important

- Se utilizzi la FleetWise console AWS IoT per creare un manifesto del decoder, AWS IoT attiva FleetWise automaticamente il manifesto del decoder per te.
- Se si utilizza l'operazione `CreateDecoderManifest` API per creare un manifesto del decodificatore, il manifesto del decodificatore rimane nello stato. **DRAFT**
- Non è possibile creare veicoli a partire da modelli di veicoli associati a un manifesto di **DRAFT** decodifica. Utilizzate l'operazione `UpdateDecoderManifest` API per modificare lo stato del manifesto del **ACTIVE** decoder.
- Non è possibile modificare i manifesti del decoder che si trovano nello stato. **ACTIVE**

Argomenti

- [Configura le interfacce di rete e i segnali del decodificatore](#)
- [Crea un manifesto del decodificatore](#)

- [Aggiornare un manifesto del decoder \(\)AWS CLI](#)
- [Eliminare un manifesto del decodificatore](#)
- [Ottieni informazioni sul manifesto del decoder \(\)AWS CLI](#)

Configura le interfacce di rete e i segnali del decodificatore

Ogni manifesto di decodifica dispone almeno di un'interfaccia di rete e di segnali di decodifica associati ai segnali specificati nel modello di veicolo associato.

Se il manifesto del decodificatore contiene interfacce di rete CAN, deve contenere segnali di decodifica CAN. Se il manifesto del decodificatore contiene interfacce di rete OBD, deve contenere segnali di decodifica OBD.

Argomenti

- [Configurare le interfacce di rete](#)
- [Configura i segnali del decoder](#)

Configurare le interfacce di rete

Per configurare un'interfaccia di rete CAN, specificare le seguenti informazioni.

- `name`— Il nome dell'interfaccia CAN.

Il nome dell'interfaccia deve essere univoco e può contenere da 1 a 100 caratteri.

- (Facoltativo)`protocolName`: il nome del protocollo.

Valori validi: CAN-FD e CAN

- (Opzionale)`protocolVersion`: l' AWS IoT FleetWise attualmente supporta CAN-FD e CAN 2.0b.

Valori validi: e 1.0 2.0b

Per configurare un'interfaccia di rete OBD, specificare le seguenti informazioni.

- `name`— Il nome dell'interfaccia OBD.

Il nome dell'interfaccia deve essere univoco e può contenere da 1 a 100 caratteri.

- `requestMessageId`— L'ID del messaggio che richiede i dati.

- (Facoltativo)`dtcRequestIntervalSeconds`: con quale frequenza richiedere i codici di errore diagnostici (DTC) al veicolo in pochi secondi. Ad esempio, se il valore specificato è 120, il software Edge Agent raccoglie i DTC memorizzati una volta ogni 2 minuti.
- (Facoltativo)`hasTransmissionEcu`: se il veicolo dispone di un modulo di controllo della trasmissione (TCM).

Valori validi: `true` e `false`

- (Opzionale)`obdStandard`: lo standard OBD FleetWise supportato dall' AWS IoT. AWS L' IoT FleetWise attualmente supporta lo standard ISO15765-4 della World Wide Harmonization On-Board Diagnostics (WWH-OBD).
- (Opzionale)`pidRequestIntervalSeconds`: con che frequenza richiedere i PID OBD II al veicolo. Ad esempio, se il valore specificato è 120, il software Edge Agent raccoglie i PID OBD II una volta ogni 2 minuti.
- (Facoltativo)`useExtendedIds`: se utilizzare gli ID estesi nel messaggio.

Valori validi: `true` e `false`

Per configurare un'interfaccia di rete middleware del veicolo, specificare le seguenti informazioni.

- `name`— Il nome dell'interfaccia middleware del veicolo.

Il nome dell'interfaccia deve essere univoco e può contenere da 1 a 100 caratteri.

- `protocolName`— Il nome del protocollo.

Valori validi: `ROS_2`

Configura i segnali del decoder

Per configurare un segnale di decodifica CAN, specificare le seguenti informazioni.

- `factor`— Il moltiplicatore utilizzato per decodificare il messaggio.
- `isBigEndian`— Se l'ordine dei byte del messaggio è big-endian. Se è big-endian, il valore più importante della sequenza viene memorizzato per primo, all'indirizzo di archiviazione più basso.
- `isSigned`— Se il messaggio è firmato. Se è firmato, il messaggio può rappresentare sia numeri positivi che negativi.
- `length`— La lunghezza del messaggio in byte.

- `messageId`— L'ID del messaggio.
- `offset`— L'offset utilizzato per calcolare il valore del segnale. Combinato con il fattore, il calcolo è $value = raw_value * factor + offset$.
- `startBit`— Indica la posizione del primo bit del messaggio.
- (Facoltativo) `name` — Il nome del segnale.

Per configurare un segnale di decodifica OBD, specificare le seguenti informazioni.

- `byteLength`— La lunghezza del messaggio in byte.
- `offset`— L'offset utilizzato per calcolare il valore del segnale. Combinato con il ridimensionamento, il calcolo è $value = raw_value * scaling + offset$.
- `pid`— Il codice diagnostico utilizzato per richiedere un messaggio da un veicolo per questo segnale.
- `pidResponseLength`— La lunghezza del messaggio richiesto.
- `scaling`— Il moltiplicatore usato per decodificare il messaggio.
- `serviceMode`— La modalità di funzionamento (servizio di diagnostica) in un messaggio.
- `startByte`— Indica l'inizio del messaggio.
- (Facoltativo) `bitMaskLength`: il numero di bit mascherati in un messaggio.
- (Facoltativo) `bitRightShift`: il numero di posizioni è stato spostato a destra.

Per configurare un segnale di decodifica dei messaggi, specificare le seguenti informazioni.

- `topicName`— Il nome dell'argomento per il segnale del messaggio. Corrisponde agli argomenti di ROS 2. Per ulteriori informazioni sull'oggetto messaggio strutturato, vedere [StructuredMessage](#).
- `structuredMessage`— Il messaggio strutturato per il segnale del messaggio. Può essere definito con `primitiveMessageDefinition`, `structuredMessageList Definition` o in modo `structuredMessageDefinition` ricorsivo.

Crea un manifesto del decodificatore

Puoi utilizzare la FleetWise console o l'API AWS IoT per creare un manifesto di decodifica per il tuo modello di veicolo.

⚠ Important

È necessario disporre di un modello di veicolo prima di poter creare un manifesto di decodifica. Ogni manifesto di decodifica deve essere associato a un modello di veicolo. Per ulteriori informazioni, consulta [Crea e gestisci modelli di veicoli](#).

Argomenti

- [Crea un manifesto del decodificatore \(console\)](#)
- [Create un manifesto del decodificatore \(\)AWS CLI](#)

Crea un manifesto del decodificatore (console)

Puoi utilizzare la FleetWise console AWS IoT per creare un manifesto di decodifica associato al tuo modello di veicolo.

⚠ Important

Non è possibile configurare i segnali dei dati del sistema di visione nei manifesti del decoder utilizzando la console AWS FleetWise IoT. Utilizza invece il. AWS CLI I dati del sistema di visione sono in versione di anteprima e sono soggetti a modifiche.

Per creare un manifesto del decodificatore

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. Nel pannello di navigazione, scegli Modelli di veicoli.
3. Scegli il modello di veicolo di destinazione.
4. Nella pagina di riepilogo del modello di veicolo, scegli Crea manifesto del decoder, quindi procedi come segue.

Argomenti

- [Passaggio 1: configura il manifesto del decoder](#)
- [Fase 2: Aggiungere interfacce di rete](#)
- [Fase 3: Revisione e creazione](#)

Passaggio 1: configura il manifesto del decoder

In Informazioni generali, procedi come segue.

1. Immettete un nome univoco per il manifesto del decodificatore.
2. (Opzionale) Immettere una descrizione.
3. Seleziona Successivo.

Fase 2: Aggiungere interfacce di rete

Ogni manifesto del decodificatore deve avere almeno un'interfaccia di rete. È possibile aggiungere più interfacce di rete a un manifesto del decodificatore.

Per aggiungere un'interfaccia di rete

- In Interfaccia di rete, effettuate le seguenti operazioni.
 - a. Per Tipo di interfaccia di rete, scegliete CAN_INTERFACE o OBD_INTERFACE.
 - b. Immettete un nome univoco per l'interfaccia di rete.
 - c. Inserisci un ID di interfaccia di rete univoco. Puoi utilizzare l'ID generato dall' AWS IoT FleetWise.
 - d. Seleziona uno o più segnali specificati nel modello del tuo veicolo da abbinare ai segnali del decoder.
 - e. Per fornire informazioni di decodifica, carica un file.dbc. AWS IoT FleetWise analizza il file.dbc per recuperare i segnali del decoder.
 - f. Nella sezione Segnali accoppiati, assicurati che ogni segnale sia associato a un segnale di decodifica.
 - g. Seleziona Successivo.

Note

- Puoi caricare solo un file.dbc per ogni interfaccia di rete.
- Assicurati che ogni segnale specificato nel modello del tuo veicolo sia associato a un segnale di decodifica.

- Dopo aver scelto di aggiungere un'altra interfaccia di rete, non puoi modificare quella che stai modificando. Puoi eliminare qualsiasi interfaccia di rete esistente.

Fase 3: Revisione e creazione

Verifica le configurazioni per il manifesto del decoder, quindi scegli Crea.

Creare un manifesto del decodificatore (AWS CLI)

È possibile utilizzare l'operazione [CreateDecoderManifest](#) API per creare manifesti del decodificatore. Gli esempi seguenti utilizzano AWS CLI.

Important

Prima di creare un manifesto del decodificatore, create innanzitutto un modello di veicolo. Per ulteriori informazioni, consulta [Crea un modello di veicolo](#).

Per creare un manifesto del decodificatore, eseguite il comando seguente.

Sostituisci *decoder-manifest-configuration* con il nome del file JSON che contiene la configurazione.

```
aws iotfleetwise create-decoder-manifest --cli-input-json file://decoder-manifest-configuration.json
```

- Sostituisci *decoder-manifest-name* con il nome del manifesto del decoder che stai creando.
- Sostituisci *Vehicle-model-ARN* con l'Amazon Resource Name (ARN) del modello di veicolo.
- (Facoltativo) Sostituisci la *descrizione con una descrizione* per aiutarti a identificare il manifesto del decoder.

Per ulteriori informazioni su come configurare rami, attributi, sensori e attuatori, vedere [Configura le interfacce di rete e i segnali del decodificatore](#)

```
{  
  "name": "decoder-manifest-name",  
  "modelManifestArn": "vehicle-model-arn",  
  "description": "description",
```

```
"networkInterfaces": [
  {
    "canInterface": {
      "name": "myNetworkInterface",
      "protocolName": "CAN",
      "protocolVersion": "2.0b"
    },
    "interfaceId": "Qq1acaenBy0B3sSM39SYm",
    "type": "CAN_INTERFACE"
  }
],
"signalDecoders": [
  {
    "canSignal": {
      "name": "Engine_Idle_Time",
      "factor": 1,
      "isBigEndian": true,
      "isSigned": false,
      "length": 24,
      "messageId": 271343712,
      "offset": 0,
      "startBit": 16
    },
    "fullyQualifiedName": "Vehicle.EngineIdleTime",
    "interfaceId": "Qq1acaenBy0B3sSM39SYm",
    "type": "CAN_SIGNAL"
  },
  {
    "canSignal": {
      "name": "Engine_Run_Time",
      "factor": 1,
      "isBigEndian": true,
      "isSigned": false,
      "length": 24,
      "messageId": 271343712,
      "offset": 0,
      "startBit": 40
    },
    "fullyQualifiedName": "Vehicle.EngineRunTime",
    "interfaceId": "Qq1acaenBy0B3sSM39SYm",
    "type": "CAN_SIGNAL"
  }
]
```

```
}
```

- *decoder-manifest-name* Sostituiscilo con il nome del manifesto del decodificatore che stai creando.
- Sostituisci *Vehicle-model-ARN* con l'Amazon Resource Name (ARN) del modello di veicolo.
- (Facoltativo) Sostituisci la *descrizione con una descrizione* per aiutarti a identificare il manifesto del decoder.

L'ordine dei nodi di proprietà all'interno di una struttura (struttura) deve rimanere coerente come definito nel catalogo dei segnali e nel modello del veicolo (manifesto del modello). Per ulteriori informazioni su come configurare rami, attributi, sensori e attuatori, vedere [Configura le interfacce di rete e i segnali del decodificatore](#).

```
{
  "name": "decoder-manifest-name",
  "modelManifestArn": "vehicle-model-arn",
  "description": "description",
  "networkInterfaces": [{
    "canInterface": {
      "name": "myNetworkInterface",
      "protocolName": "CAN",
      "protocolVersion": "2.0b"
    },
    "interfaceId": "Qq1acaenBy0B3sSM39SYm",
    "type": "CAN_INTERFACE"
  }, {
    "type": "VEHICLE_MIDDLEWARE",
    "interfaceId": "G1KzxkdnmV5Hn7wkV3ZL9",
    "vehicleMiddleware": {
      "name": "ROS2_test",
      "protocolName": "ROS_2"
    }
  }
  ]],
  "signalDecoders": [{
    "canSignal": {
      "name": "Engine_Idle_Time",
      "factor": 1,
      "isBigEndian": true,
      "isSigned": false,
      "length": 24,
```

```

    "messageId": 271343712,
    "offset": 0,
    "startBit": 16
  },
  "fullyQualifiedNames": ["Vehicle.EngineIdleTime"],
  "interfaceId": "Qq1acaenBy0B3sSM39SYm",
  "type": "CAN_SIGNAL"
},
{
  "canSignal": {
    "name": "Engine_Run_Time",
    "factor": 1,
    "isBigEndian": true,
    "isSigned": false,
    "length": 24,
    "messageId": 271343712,
    "offset": 0,
    "startBit": 40
  },
  "fullyQualifiedNames": ["Vehicle.EngineRunTime"],
  "interfaceId": "Qq1acaenBy0B3sSM39SYm",
  "type": "CAN_SIGNAL"
},
{
  "fullyQualifiedNames": ["Vehicle.CompressedImageTopic"],
  "type": "MESSAGE_SIGNAL",
  "interfaceId": "G1KzxkdnmV5Hn7wkV3ZL9",
  "messageSignal": {
    "topicName": "CompressedImageTopic:sensor_msgs/msg/CompressedImage",
    "structuredMessage": {
      "structuredMessageDefinition": [{
        "fieldName": "header",
        "dataType": {
          "structuredMessageDefinition": [{
            "fieldName": "stamp",
            "dataType": {
              "structuredMessageDefinition": [{
                "fieldName": "sec",
                "dataType": {
                  "primitiveMessageDefinition": {
                    "ros2PrimitiveMessageDefinition": {
                      "primitiveType": "INT32"
                    }
                  }
                }
              ]
            }
          ]
        }
      ]
    }
  }
}

```

```
    }
  },
  {
    "fieldName": "nanosec",
    "dataType": {
      "primitiveMessageDefinition": {
        "ros2PrimitiveMessageDefinition": {
          "primitiveType": "UINT32"
        }
      }
    }
  }
]
},
{
  "fieldName": "frame_id",
  "dataType": {
    "primitiveMessageDefinition": {
      "ros2PrimitiveMessageDefinition": {
        "primitiveType": "STRING"
      }
    }
  }
}
],
},
{
  "fieldName": "format",
  "dataType": {
    "primitiveMessageDefinition": {
      "ros2PrimitiveMessageDefinition": {
        "primitiveType": "STRING"
      }
    }
  }
}
},
{
  "fieldName": "data",
  "dataType": {
    "structuredMessageListDefinition": {
      "name": "listType",
      "memberType": {
```


⚠ Important

Dopo aver attivato il manifesto del decoder, non è possibile modificarlo.

Eliminare un manifesto del decodificatore

Puoi utilizzare la FleetWise console o l'API AWS IoT per eliminare un manifesto del decodificatore.

⚠ Important

I veicoli associati al manifesto del decoder devono essere prima eliminati. Per ulteriori informazioni, consulta [Eliminare un veicolo](#).

Argomenti

- [Eliminare un manifesto del decodificatore \(console\)](#)
- [Eliminare un manifesto del decodificatore \(\)AWS CLI](#)

Eliminare un manifesto del decodificatore (console)

È possibile utilizzare la FleetWise console AWS IoT per eliminare un manifesto del decoder.

Per eliminare un manifesto del decodificatore

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. Nel pannello di navigazione, scegli Modelli di veicoli.
3. Scegli il modello di veicolo di destinazione.
4. Nella pagina di riepilogo del modello di veicolo, scegli la scheda Manifesti del decodificatore.
5. Scegli il manifesto del decoder di destinazione, quindi scegli Elimina.
6. In Elimina **decoder-manifest-name?**, immettete il nome del manifesto del decodificatore da eliminare, quindi scegliete Conferma.

Eliminare un manifesto del decodificatore ()AWS CLI

È possibile utilizzare l'operazione [DeleteDecoderManifest](#)API per eliminare un manifesto del decodificatore. L'esempio seguente utilizza AWS CLI.

⚠ Important

Prima di eliminare il manifesto del decodificatore, eliminate prima i veicoli associati. Per ulteriori informazioni, consulta [Eliminare un veicolo](#).

Per eliminare un manifesto del decodificatore, eseguite il comando seguente.

Sostituiscilo *decoder-manifest-name* con il nome del manifesto del decoder che stai eliminando.

```
aws iotfleetwise delete-decoder-manifest --name decoder-manifest-name
```

Ottieni informazioni sul manifesto del decoder ()AWS CLI

È possibile utilizzare l'operazione [ListDecoderManifests](#) API per verificare se un manifesto del decodificatore è stato eliminato. L'esempio seguente utilizza AWS CLI.

Per recuperare un elenco impaginato di riepiloghi di tutti i manifesti del decodificatore, eseguite il comando seguente.

```
aws iotfleetwise list-decoder-manifests
```

È possibile utilizzare l'operazione [ListDecoderManifestSignals](#) API per verificare se i segnali del decoder nel manifesto del decoder sono stati aggiornati. L'esempio seguente utilizza AWS CLI.

Per recuperare un elenco impaginato di riepiloghi di tutti i segnali del decodificatore (nodi) in un determinato manifesto del decodificatore, eseguite il comando seguente.

Sostituiscilo *decoder-manifest-name* con il nome del manifesto del decodificatore che stai controllando.

```
aws iotfleetwise list-decoder-manifest-signals /  
--name decoder-manifest-name
```

È possibile utilizzare l'operazione [ListDecoderManifestNetworkInterfaces](#) API per verificare se le interfacce di rete nel manifesto del decoder sono state aggiornate. Nell'esempio seguente viene utilizzato AWS CLI.

Per recuperare un elenco impaginato di riepiloghi di tutte le interfacce di rete in un determinato manifesto del decoder, eseguite il comando seguente.

Sostituiscilo *decoder-manifest-name* con il nome del manifesto del decodificatore che stai controllando.

```
aws iotfleetwise list-decoder-manifest-network-interfaces /  
    --name decoder-manifest-name
```

È possibile utilizzare l'operazione [GetDecoderManifest](#) API per verificare se le interfacce di rete e i segnali di decodifica nel manifesto del decoder sono stati aggiornati. L'esempio seguente utilizza AWS CLI

Per recuperare informazioni su un manifesto del decodificatore, eseguite il comando seguente.

Sostituite *decoder-manifest* con il nome del decoder manifest che desiderate recuperare.

```
aws iotfleetwise get-decoder-manifest --name decoder-manifest
```

Note

Questa operazione è [consistente finale](#). In altre parole, le modifiche al manifesto del decoder potrebbero non riflettersi immediatamente.

Crea, fornisci e gestisci veicoli

I veicoli sono esempi di modelli di veicoli. I veicoli devono essere creati a partire da un modello di veicolo e associati a un manifesto di decodifica. I veicoli caricano uno o più flussi di dati sul cloud. Ad esempio, un veicolo può inviare al cloud i dati su chilometraggio, temperatura del motore e stato del riscaldatore. Ogni veicolo contiene le seguenti informazioni:

`vehicleName`

Un ID che identifica il veicolo.

Non aggiungere informazioni di identificazione personale (PII) o altre informazioni riservate o sensibili nel nome del veicolo. I nomi dei veicoli sono accessibili tramite altri AWS servizi, tra cui Amazon CloudWatch. I nomi dei veicoli non sono pensati per essere utilizzati per dati privati o sensibili.

`modelManifestARN`

L'Amazon Resource Name (ARN) di un modello di veicolo (manifesto del modello). Ogni veicolo viene creato a partire da un modello di veicolo. I veicoli creati a partire dallo stesso modello di veicolo sono costituiti dallo stesso gruppo di segnali ereditati dal modello di veicolo. Questi segnali sono definiti e standardizzati nel catalogo dei segnali.

`decoderManifestArn`

L'ARN del manifesto del decoder. Un manifesto di decodifica fornisce informazioni di decodifica che l'AWS IoT FleetWise può utilizzare per trasformare i dati grezzi del segnale (dati binari) in valori leggibili dall'uomo. Un manifesto del decodificatore deve essere associato a un modello di veicolo. AWS IoT FleetWise utilizza lo stesso manifesto di decodifica per decodificare i dati grezzi dei veicoli creati sulla base dello stesso modello di veicolo.

`attributes`

Gli attributi sono coppie chiave-valore che contengono informazioni statiche. I veicoli possono contenere attributi ereditati dal modello di veicolo. Puoi aggiungere attributi aggiuntivi per distinguere un singolo veicolo dagli altri veicoli creati dallo stesso modello di veicolo. Per esempio, se hai un'auto nera, puoi specificare il seguente valore per un attributo: `{"color": "black"}`.

⚠ Important

Gli attributi devono essere definiti nel modello di veicolo associato prima di poterli aggiungere ai singoli veicoli.

Per ulteriori informazioni sui modelli di veicoli, i manifesti dei decoder e gli attributi, vedere.

[Modellazione di veicoli](#)

AWS L'IoT FleetWise fornisce le seguenti operazioni API che puoi utilizzare per creare e gestire veicoli.

- [CreateVehicle](#)— Crea un nuovo veicolo.
- [BatchCreateVehicle](#)— Crea uno o più veicoli nuovi.
- [UpdateVehicle](#)— Aggiorna un veicolo esistente.
- [BatchUpdateVehicle](#)— Aggiorna uno o più veicoli esistenti.
- [DeleteVehicle](#)— Elimina un veicolo esistente.
- [ListVehicles](#)— Recupera un elenco impaginato di riepiloghi di tutti i veicoli.
- [GetVehicle](#)— Recupera informazioni su un veicolo.

Tutorial

- [Veicoli di approvvigionamento](#)
- [Argomenti riservati](#)
- [Crea un veicolo](#)
- [Aggiorna un veicolo \(AWS CLI\)](#)
- [Aggiorna più veicoli \(AWS CLI\)](#)
- [Eliminare un veicolo](#)
- [Ottieni informazioni sul veicolo \(AWS CLI\)](#)

Veicoli di approvvigionamento

Il FleetWise software Edge Agent for AWS IoT in esecuzione nel veicolo raccoglie e trasferisce i dati nel cloud. AWS L'IoT FleetWise si integra AWS IoT Core per supportare una comunicazione sicura tra il software Edge Agent e il cloud tramite MQTT. Ogni veicolo corrisponde a qualcosa. AWS IoT

Puoi usare un AWS IoT oggetto esistente per creare un veicolo o impostare l' AWS IoT FleetWise per creare automaticamente un AWS IoT oggetto per il tuo veicolo. Per ulteriori informazioni, consulta [Crea un veicolo \(AWS CLI\)](#).

AWS IoT Core supporta [l'autenticazione e l'autorizzazione](#) che aiutano a controllare in modo sicuro l'accesso alle FleetWise risorse AWS IoT. I veicoli possono utilizzare i certificati X.509 per autenticarsi (accedere) per utilizzare l' AWS IoT FleetWise e AWS IoT Core le policy per ottenere l'autorizzazione (disporre delle autorizzazioni) per eseguire azioni specifiche.

Autentica i veicoli

Puoi creare AWS IoT Core politiche per autenticare i tuoi veicoli.

Per autenticare il tuo veicolo

- Per creare una AWS IoT Core politica, esegui il comando seguente.
 - Sostituisci *policy-name* con il nome della politica che desideri creare.
 - Sostituisci *file-name* con il nome del file JSON che contiene la policy. AWS IoT Core

```
aws iot create-policy --policy-name policy-name --policy-document file://file-name.json
```

Prima di utilizzare la politica di esempio, procedi come segue:

- Sostituisci la *regione* con la AWS regione in cui hai creato FleetWise le risorse AWS IoT.
- Sostituisci *AWSAccount* con l'ID del tuo AWS account.

Questo esempio include argomenti riservati all' AWS IoT FleetWise. È necessario aggiungere gli argomenti alla politica. Per ulteriori informazioni, consulta [Argomenti riservati](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:Connect"
      ]
    }
  ]
}
```

```

    ],
    "Resource": [
        "arn:aws:iot:region:awsAccount:client/
${iot:Connection.Thing.ThingName}"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "iot:Publish"
    ],
    "Resource": [
        "arn:aws:iot:region:awsAccount:topic/$aws/iotfleetwise/vehicles/
${iot:Connection.Thing.ThingName}/checkins",
        "arn:aws:iot:region:awsAccount:topic/$aws/iotfleetwise/vehicles/
${iot:Connection.Thing.ThingName}/signals"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "iot:Subscribe"
    ],
    "Resource": [
        "arn:aws:iot:region:awsAccount:topicfilter/$aws/iotfleetwise/
vehicles/${iot:Connection.Thing.ThingName}/collection_schemes",
        "arn:aws:iot:region:awsAccount:topicfilter/$aws/iotfleetwise/
vehicles/${iot:Connection.Thing.ThingName}/decoder_manifests"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "iot:Receive"
    ],
    "Resource": [
        "arn:aws:iot:region:awsAccount:topic/$aws/iotfleetwise/vehicles/
${iot:Connection.Thing.ThingName}/collection_schemes",
        "arn:aws:iot:region:awsAccount:topic/$aws/iotfleetwise/vehicles/
${iot:Connection.Thing.ThingName}/decoder_manifests"
    ]
}
]

```

```
}
```

Autorizza i veicoli

Puoi creare certificati X.509 per autorizzare i tuoi veicoli.

Per autorizzare il tuo veicolo

Important

Ti consigliamo di creare un nuovo certificato per ogni veicolo.

1. Per creare una coppia di key pair RSA ed emettere un certificato X.509, esegui il comando seguente.
 - Sostituisci *cert* con il nome del file che salva il contenuto dell'output del comando di CertificatePEM.
 - Sostituisci *public-key* con il nome del file che salva il contenuto dell'output del comando di KeyPair. PublicKey.
 - Sostituisci la *chiave privata* con il nome del file che salva il contenuto dell'output del comando di KeyPair. PrivateKey.

```
aws iot create-keys-and-certificate \  
  --set-as-active \  
  --certificate-pem-outfile cert.pem \  
  --public-key-outfile public-key.key" \  
  --private-key-outfile private-key.key"
```

2. Copia l'Amazon Resource Name (ARN) del certificato dall'output.
3. Per allegare la policy al certificato, esegui il comando seguente.
 - Sostituisci *policy-name* con il nome della AWS IoT Core politica che hai creato.
 - Sostituisci *certificate-arn* con l'ARN del certificato che hai copiato.

```
aws iot attach-policy \  
  --policy-name policy-name\
```

```
--target "certificate-arn"
```

4. Per allegare il certificato all'oggetto, esegui il comando seguente.

- Sostituisci *thing-name* con il nome dell' AWS IoT oggetto o l'ID del veicolo.
- Sostituisci *certificate-arn* con l'ARN del certificato che hai copiato.

```
aws iot attach-thing-principal \  
  --thing-name thing-name \  
  --principal "certificate-arn"
```

Argomenti riservati

AWS L'IoT FleetWise si riserva l'uso dei seguenti argomenti. Se l'argomento riservato lo consente, puoi abbonarti o pubblicarlo. Tuttavia, non puoi creare nuovi argomenti che iniziano con il simbolo del dollaro (\$). Se utilizzi operazioni di pubblicazione o sottoscrizione non supportate con argomenti riservati, la connessione può terminare.

Argomento	Operazione del client consentita	Descrizione
\$aws/iotfleetwise/vehicles/ <i>vehicleName</i> /checkins	Pubblicare	<p>Il software Edge Agent pubblica informazioni sullo stato del veicolo su questo argomento.</p> <p>Le informazioni sullo stato del veicolo vengono scambiate in formato protobuf (protobuf). Per ulteriori informazioni, consulta la Guida per gli sviluppatori del FleetWise software</p>

Argomento	Operazione del client consentita	Descrizione	
		Edge Agent for AWS IoT.	
<code>\$aws/iotfleetwise/vehicles/<i>vehicleName</i> /signals</code>	Pubblicare	<p>Il software Edge Agent pubblica segnali relativi a questo argomento.</p> <p>Le informazioni sui segnali vengono scambiate in formato buffer di protocollo (protobuf). Per ulteriori informazioni, consulta la Guida per gli sviluppatori del FleetWise software Edge Agent for AWS IoT.</p>	
<code>\$aws/iotfleetwise/vehicles/<i>vehicleName</i> /collection_schemes</code>	Subscribe	<p>AWS IoT FleetWise pubblica schemi di raccolta dati su questo argomento. I veicoli utilizzano questi schemi di raccolta dati.</p>	
<code>\$aws/iotfleetwise/vehicles/<i>vehicleName</i> /decoder_manifests</code>	Subscribe	<p>AWS IoT FleetWise pubblica i manifesti dei decoder su questo argomento. I veicoli utilizzano questi manifesti di decodifica.</p>	

Crea un veicolo

Puoi utilizzare la FleetWise console o l'API AWS IoT per creare un veicolo.

Important

Prima di iniziare, controlla quanto segue:

- È necessario disporre di un modello di veicolo e lo stato del modello di veicolo deve essere ACTIVE. Per ulteriori informazioni, consulta [Crea e gestisci modelli di veicoli](#).
- Il modello del veicolo deve essere associato a un manifesto del decodificatore e lo stato del manifesto del decodificatore deve essere ACTIVE. Per ulteriori informazioni, consulta [Crea e gestisci i manifesti del decodificatore](#).

Argomenti

- [Crea un veicolo \(console\)](#)
- [Crea un veicolo \(AWS CLI\)](#)
- [Crea più veicoli \(AWS CLI\)](#)

Crea un veicolo (console)

Puoi utilizzare la FleetWise console AWS IoT per creare un veicolo.

Important

Prima di iniziare, controlla quanto segue:

- È necessario disporre di un modello di veicolo e lo stato del modello di veicolo deve essere ACTIVE. Per ulteriori informazioni, consulta [Crea e gestisci modelli di veicoli](#).
- Il modello del veicolo deve essere associato a un manifesto del decodificatore e lo stato del manifesto del decodificatore deve essere ACTIVE. Per ulteriori informazioni, consulta [Crea e gestisci i manifesti del decodificatore](#).

Per creare un veicolo

1. Apri la [FleetWise console AWS IoT](#).
2. Nel pannello di navigazione, scegli Veicoli.
3. Nella pagina di riepilogo del veicolo, scegli Crea veicolo, quindi procedi nel seguente modo.

Argomenti

- [Fase 1: Definizione delle proprietà del veicolo](#)
- [Fase 2: Configurare il certificato del veicolo](#)
- [Fase 3: Allega le politiche al certificato](#)
- [Fase 4: Revisione e creazione](#)

Fase 1: Definizione delle proprietà del veicolo

In questo passaggio, si assegna un nome al veicolo e lo si associa al manifesto del modello e al manifesto del decodificatore.

1. Inserite un nome univoco per il veicolo.

Important

Un veicolo corrisponde a AWS IoT qualcosa. Se esiste già un oggetto con quel nome, scegli Associa il veicolo a un oggetto IoT per aggiornare l'oggetto al veicolo. Oppure, scegli un nome diverso per il veicolo e l'AWS IoT FleetWise creerà automaticamente qualcosa di nuovo per il veicolo.

2. Scegli un modello di veicolo (manifesto del modello) dall'elenco.
3. Scegliete un manifesto del decodificatore dall'elenco. Il manifesto del decodificatore è associato al modello del veicolo.
4. (Facoltativo) Per associare gli attributi del veicolo, scegliete Aggiungi attributi. Se salti questo passaggio, devi aggiungere gli attributi dopo la creazione del veicolo prima di poterlo distribuire nelle campagne.
5. (Facoltativo) Per associare i tag al veicolo, scegli Aggiungi nuovo tag. Puoi anche aggiungere tag dopo aver creato il veicolo.
6. Seleziona Successivo.

Fase 2: Configurare il certificato del veicolo

Per utilizzare il veicolo come AWS IoT oggetto, è necessario configurare un certificato del veicolo con una politica allegata. Se salti questo passaggio, devi configurare un certificato dopo la creazione del veicolo prima di poterlo distribuire nelle campagne.

1. Scegli Genera automaticamente un nuovo certificato (consigliato).
2. Seleziona Successivo.

Fase 3: Allega le politiche al certificato

Allega una policy al certificato che hai configurato nel passaggio precedente.

1. Per Politiche, inserisci il nome di una politica esistente. Per creare una nuova politica, scegli Crea politica.
2. Seleziona Successivo.

Fase 4: Revisione e creazione

Verifica le configurazioni per il veicolo, quindi scegli Crea veicolo.

Important

Dopo aver creato il veicolo, devi scaricare il certificato e le chiavi. Utilizzerai il certificato e la chiave privata per connettere il veicolo nel FleetWise software Edge Agent for AWS IoT.

Crea un veicolo (AWS CLI)

Quando si crea un veicolo, è necessario utilizzare un modello di veicolo associato a un manifesto di decodifica. È possibile utilizzare l'operazione [CreateVehicleAPI](#) per creare un veicolo. Gli esempi seguenti utilizzano AWS CLI.

Important

Prima di iniziare, controlla quanto segue:

- È necessario disporre di un modello di veicolo e lo stato del modello di veicolo deve essere ACTIVE. Per ulteriori informazioni, consulta [Crea e gestisci modelli di veicoli](#).

- Il modello del veicolo deve essere associato a un manifesto del decodificatore e lo stato del manifesto del decodificatore deve essere. ACTIVE Per ulteriori informazioni, consulta [Crea e gestisci i manifesti del decodificatore](#).

Per creare un veicolo, esegui il comando seguente.

Sostituisci *file-name* con il nome del file JSON che contiene la configurazione del veicolo.

```
aws iotfleetwise create-vehicle --cli-input-json file://file-name.json
```

Example configurazione del veicolo

- (Facoltativo) Il `associationBehavior` valore può essere uno dei seguenti:
 - `CreateIotThing`— Quando il veicolo viene creato, l' AWS IoT crea FleetWise automaticamente un AWS IoT elemento con il nome dell'ID del veicolo.
 - `ValidateIotThingExists`— Usa un AWS IoT oggetto esistente per creare un veicolo.

Per creare AWS IoT qualcosa, esegui il seguente comando. Sostituisci *thing-name* con il nome dell'oggetto che desideri creare.

```
aws iot create-thing --thing-name thing-name
```

Se non è specificato, l' AWS IoT crea FleetWise automaticamente AWS IoT qualcosa per il tuo veicolo.

Important

Assicurati che l' AWS IoT oggetto venga fornito dopo la creazione del veicolo. Per ulteriori informazioni, consulta [Veicoli di approvvigionamento](#).

- Sostituisci il *nome del veicolo* con uno dei seguenti.
 - Il nome della AWS IoT cosa se `associationBehavior` è configurato su `ValidateIotThingExists`
 - L'ID del veicolo su cui creare if `associationBehavior` è configurato `CreateIotThing`.

L'ID del veicolo può contenere da 1 a 100 caratteri. Caratteri validi: a—z, A—Z, 0—9, trattino (-), trattino basso (_) e due punti (:).

- Sostituisci *model-manifest-ARN* con l'ARN del tuo modello di veicolo (model manifest).
- Sostituisci *decoder-manifest-ARN* con l'ARN del manifesto del decoder associato al modello di veicolo specificato.
- (Facoltativo) Puoi aggiungere attributi aggiuntivi per distinguere questo veicolo dagli altri veicoli creati dallo stesso modello di veicolo. Ad esempio, se hai un'auto elettrica, puoi specificare il seguente valore per un attributo:{"fuelType": "electric"}.

⚠ Important

Gli attributi devono essere definiti nel modello di veicolo associato prima di poterli aggiungere ai singoli veicoli.

```
{
  "associationBehavior": "associationBehavior",
  "vehicleName": "vehicle-name",
  "modelManifestArn": "model-manifest-ARN",
  "decoderManifestArn": "decoder-manifest-ARN",
  "attributes": {
    "key": "value"
  }
}
```

Crea più veicoli (AWS CLI)

Puoi utilizzare l'operazione [BatchCreateVehicle](#) API per creare più veicoli contemporaneamente. Gli esempi seguenti utilizzano AWS CLI.

Per creare più veicoli, esegui il comando seguente.

Sostituisci *file-name* con il nome del file JSON che contiene le configurazioni di più veicoli.

```
aws iotfleetwise batch-create-vehicle --cli-input-json file://file-name.json
```

Example configurazioni dei veicoli

```
{
  "vehicles": [
    {
      "associationBehavior": "associationBehavior",
```

```

        "vehicleName": "vehicle-name",
        "modelManifestArn": "model-manifest-ARN",
        "decoderManifestArn": "decoder-manifest-ARN",
        "attributes": {
            "key": "value"
        }
    },
    {
        "associationBehavior": "associationBehavior",
        "vehicleName": "vehicle-name",
        "modelManifestArn": "model-manifest-ARN",
        "decoderManifestArn": "decoder-manifest-ARN",
        "attributes": {
            "key": "value"
        }
    }
]
}

```

È possibile creare fino a 10 veicoli per ogni operazione in batch. Per ulteriori informazioni sulla configurazione del veicolo, consulta [Crea un veicolo \(AWS CLI\)](#).

Aggiorna un veicolo (AWS CLI)

Puoi utilizzare l'operazione [UpdateVehicleAPI](#) per aggiornare un veicolo esistente. Gli esempi seguenti utilizzano AWS CLI.

Per aggiornare un veicolo, esegui il comando seguente.

Sostituisci *file-name* con il nome del file JSON che contiene la configurazione del tuo veicolo.

```
aws iotfleetwise update-vehicle --cli-input-json file://file-name.json
```

Example configurazione del veicolo

- Sostituisci *il nome del veicolo* con l'ID del veicolo che desideri aggiornare.
- (Facoltativo) Sostituisci *Model-manifest-ARN* con l'ARN del modello di veicolo (manifesto del modello) che utilizzi per sostituire il modello di veicolo in uso.
- (Facoltativo) Sostituisci *Decoder-Manifest-ARN* con l'ARN del tuo manifesto di decodifica associato al nuovo modello di veicolo che hai specificato.
- (Facoltativo) Sostituiscilo con gli attributi del veicolo. *attribute-update-mode*

- **Merge**— Unisce nuovi attributi in attributi esistenti aggiornando gli attributi esistenti con nuovi valori e aggiungendo nuovi attributi se non esistono.

Per esempio, se un veicolo ha i seguenti attributi:{"color": "black", "fuelType": "electric"}, e aggiorni il veicolo con i seguenti attributi:{"color": "", "fuelType": "gasoline", "model": "x"}, il veicolo aggiornato ha i seguenti attributi:{"fuelType": "gasoline", "model": "x"}.

- **Overwrite**— Sostituire gli attributi esistenti con nuovi attributi.

Ad esempio, se un veicolo ha i seguenti attributi: e {"color": "black", "fuelType": "electric"} si aggiorna il veicolo con l'{"model": "x"}attributo, il veicolo aggiornato dispone dell'{"model": "x"}attributo.

Questo è necessario se nell'input sono presenti degli attributi.

- (Facoltativo) Per aggiungere nuovi attributi o aggiornare quelli esistenti con nuovi valori, configura `attributes`. Ad esempio, se hai un'auto elettrica, puoi specificare il seguente valore per un attributo:{"fuelType": "electric"}.

Per eliminare gli attributi, configura `attributeUpdateMode` su `Merge`.

Important

Gli attributi devono essere definiti nel modello di veicolo associato prima di poterli aggiungere ai singoli veicoli.

```
{
  "vehicleName": "vehicle-name",
  "modelManifestArn": "model-manifest-arn",
  "decoderManifestArn": "decoder-manifest-arn",
  "attributeUpdateMode": "attribute-update-mode"
}
```

Aggiorna più veicoli (AWS CLI)

Puoi utilizzare l'operazione [BatchUpdateVehicleAPI](#) per aggiornare più veicoli esistenti contemporaneamente. Gli esempi seguenti utilizzano AWS CLI.

Per aggiornare più veicoli, esegui il comando seguente.

Sostituisci *file-name* con il nome del file JSON che contiene le configurazioni di più veicoli.

```
aws iotfleetwise batch-update-vehicle --cli-input-json file://file-name.json
```

Example configurazioni dei veicoli

```
{
  "vehicles": [
    {
      "vehicleName": "vehicle-name",
      "modelManifestArn": "model-manifest-arn",
      "decoderManifestArn": "decoder-manifest-arn",
      "mergeAttributes": true,
      "attributes": {
        "key": "value"
      }
    },
    {
      "vehicleName": "vehicle-name",
      "modelManifestArn": "model-manifest-arn",
      "decoderManifestArn": "decoder-manifest-arn",
      "mergeAttributes": true,
      "attributes": {
        "key": "value"
      }
    }
  ]
}
```

Puoi aggiornare fino a 10 veicoli per ogni operazione in batch. Per ulteriori informazioni sulla configurazione di ciascun veicolo, consulta [Aggiorna un veicolo \(AWS CLI\)](#).

Eliminare un veicolo

Puoi utilizzare la FleetWise console o l'API AWS IoT per eliminare i veicoli.

Important

Dopo l'eliminazione di un veicolo, l'AWS IoT FleetWise lo rimuove automaticamente dalle flotte e dalle campagne associate. Per ulteriori informazioni, consulta [Crea e gestisci flotte](#) e

[Raccogli e trasferisci dati con le campagne](#). Tuttavia, il veicolo esiste ancora come oggetto o è ancora associato a qualcosa dentro AWS IoT Core. Per istruzioni sull'eliminazione di un elemento, consulta [Eliminare un oggetto](#) nella Guida per gli AWS IoT Core sviluppatori.

Eliminare un veicolo (console)

Puoi utilizzare la FleetWise console AWS IoT per eliminare un veicolo.

Per eliminare un veicolo

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. Nel pannello di navigazione, scegli Veicoli.
3. Nella pagina Veicoli, seleziona il pulsante accanto al veicolo che desideri eliminare.
4. Scegli Elimina.
5. In Elimina **vehicle-name**, inserisci il nome del veicolo, quindi scegli Elimina.

Elimina un veicolo (AWS CLI)

Puoi utilizzare l'operazione [DeleteVehicle](#)API per eliminare un veicolo. L'esempio seguente utilizza AWS CLI.

Per eliminare un veicolo, esegui il comando seguente.

Sostituisci *il nome del veicolo* con l'ID del veicolo che desideri eliminare.

```
aws iotfleetwise delete-vehicle --vehicle-name vehicle-name
```

Ottieni informazioni sul veicolo ()AWS CLI

Puoi utilizzare l'operazione [ListVehicles](#)API per verificare se un veicolo è stato eliminato. Gli esempi seguenti utilizzano AWS CLI.

Per recuperare un elenco impaginato di riepiloghi di tutti i veicoli, esegui il comando seguente.

```
aws iotfleetwise list-vehicles
```

È possibile utilizzare l'operazione [GetVehicle](#) API per recuperare le informazioni sul veicolo. Gli esempi seguenti utilizzano AWS CLI.

Per recuperare i metadati di un veicolo, esegui il comando seguente.

Sostituisci *il nome del veicolo* con l'ID del veicolo che desideri recuperare.

```
aws iotfleetwise get-vehicle --vehicle-name vehicle-name
```

Note

Questa operazione è [consistente finale](#). In altre parole, le modifiche al veicolo potrebbero non riflettersi immediatamente.

Crea e gestisci flotte

Una flotta rappresenta un gruppo di veicoli. Una flotta senza veicoli associati è un'entità vuota. Prima di poter utilizzare la flotta per gestire più veicoli contemporaneamente, è necessario associare i veicoli alla flotta. Un veicolo può appartenere a più istanze. Puoi controllare quali dati raccogliere da una flotta di veicoli e quando raccogliere i dati implementando una campagna. Per ulteriori informazioni, consulta [Raccogli e trasferisci dati con le campagne](#).

Una flotta contiene le seguenti informazioni.

`fleetId`

L'ID della flotta.

(Facoltativo) `description`

Una descrizione che ti aiuta a trovare la flotta.

`signalCatalogArn`

L'Amazon Resource Name (ARN) del catalogo di segnali.

AWS IoT FleetWise fornisce le seguenti operazioni API che puoi utilizzare per creare e gestire flotte.

- [CreateFleet](#)— Crea un gruppo di veicoli che contengono lo stesso gruppo di segnali.
- [AssociateVehicleFleet](#)— Associa un veicolo a una flotta.
- [DisassociateVehicleFleet](#)— Dissocia un veicolo da una flotta.
- [UpdateFleet](#)— Aggiorna la descrizione di una flotta esistente.
- [DeleteFleet](#)— Elimina una flotta esistente.
- [ListFleets](#)— Recupera un elenco impaginato di riepiloghi di tutte le flotte.
- [ListFleetsForVehicle](#)— Recupera un elenco impaginato di ID di tutte le flotte a cui appartiene il veicolo.
- [ListVehiclesInFleet](#)— Recupera un elenco impaginato di riepiloghi di tutti i veicoli di una flotta.
- [GetFleet](#)— Recupera informazioni su una flotta.

Argomenti

- [Crea una flotta \(AWS CLI\)](#)

- [Associa un veicolo a una flotta \(AWS CLI\)](#)
- [Dissociare un veicolo da una flotta \(AWS CLI\)](#)
- [Aggiorna una flotta \(AWS CLI\)](#)
- [Elimina un parco istanze \(AWS CLI\)](#)
- [Ottieni informazioni sulla flotta \(AWS CLI\)](#)

Crea una flotta (AWS CLI)

Puoi utilizzare l'operazione [CreateFleet](#) API per creare una flotta di veicoli. Nell'esempio seguente viene utilizzato AWS CLI.

Important

Per poter creare una flotta, è necessario disporre di un catalogo di segnali. Per ulteriori informazioni, consulta [Crea un catalogo di segnali \(AWS CLI\)](#).

Per creare una flotta, eseguire questo comando riportato di seguito.

- *Sostituisci l'id* del parco veicoli con l'ID del parco veicoli che stai creando.

L'ID della flotta deve essere univoco e contenere da 1 a 100 caratteri. Caratteri validi: lettere (A-Z), numeri (0-9), due punti (:), trattini (-) e caratteri di sottolineatura (_).

- (Facoltativo) Sostituisci la *descrizione* con una descrizione.

La descrizione può contenere 1-2048 caratteri.

- Sostituire *signal-catalog-arn* con l'ARN del catalogo di segnali.

```
aws iotfleetwise create-fleet \  
  --fleet-id fleet-id \  
  --description description \  
  --signal-catalog-arn signal-catalog-arn
```

Associa un veicolo a una flotta (AWS CLI)

Puoi utilizzare l'operazione [AssociateVehicleFleet](#) API per associare un veicolo a una flotta. Nell'esempio seguente viene utilizzato AWS CLI.

Important

- È necessario disporre di un veicolo e di una flotta prima di poter associare un veicolo a una flotta. Per ulteriori informazioni, consulta [Crea, fornisci e gestisci veicoli](#).
- Se associ un veicolo a una flotta oggetto di una campagna, l'AWS IoT distribuisce FleetWise automaticamente la campagna sul veicolo.

Per associare un veicolo a una flotta, eseguire questo comando riportato di seguito.

- Sostituisci l'*ID della flotta* con l'ID della flotta.
- Sostituisci *il nome del veicolo* con l'ID del veicolo.

```
aws iotfleetwise associate-vehicle-fleet --fleet-id fleet-id --vehicle-name vehicle-name
```

Dissociare un veicolo da una flotta (AWS CLI)

Puoi utilizzare l'operazione [DisassociateVehicleFleet](#) API per dissociare un veicolo da una flotta. Nell'esempio seguente viene utilizzato AWS CLI.

Per dissociare un veicolo da una flotta, eseguire questo comando riportato di seguito.

- Sostituisci l'*ID della flotta* con l'ID della flotta.
- Sostituisci *il nome del veicolo* con l'ID del veicolo.

```
aws iotfleetwise disassociate-vehicle-fleet --fleet-id fleet-id --vehicle-name vehicle-name
```

Aggiorna una flotta (AWS CLI)

È possibile utilizzare l'operazione [UpdateFleet](#) API per aggiornare la descrizione di un parco veicoli. Nell'esempio seguente viene utilizzato AWS CLI.

Per aggiornare un parco istanze, eseguire questo comando riportato di seguito.

- Sostituisci l'*ID della flotta* con l'ID della flotta che stai aggiornando.
- Sostituisci la *descrizione* con una nuova descrizione.

La descrizione può contenere 1-2048 caratteri.

```
aws iotfleetwise update-fleet --fleet-id fleet-id --description description
```

Elimina un parco istanze (AWS CLI)

È possibile utilizzare l'operazione [DeleteFleet](#) API per eliminare un parco veicoli. Nell'esempio seguente viene utilizzato AWS CLI.

Important

Prima di eliminare una flotta, assicurati che non abbia veicoli associati. Per istruzioni su come dissociare un veicolo da una flotta, consulta [Dissociare un veicolo da una flotta \(AWS CLI\)](#).

Per eliminare un parco istanze, eseguire questo comando riportato di seguito.

Sostituisci l'*ID* della flotta con l'ID del parco veicoli che stai eliminando.

```
aws iotfleetwise delete-fleet --fleet-id fleet-id
```

Ottieni informazioni sulla flotta (AWS CLI)

È possibile utilizzare l'operazione [ListFleets](#) API per verificare se un parco veicoli è stato eliminato. Gli esempi seguenti utilizzano AWS CLI.

Per recuperare un elenco di riepiloghi in pagine di riepiloghi di riepiloghi di diversi istanze, eseguire il comando riportato di seguito.

```
aws iotfleetwise list-fleets
```

È possibile utilizzare l'operazione [ListFleetsForVehicle](#) API per recuperare un elenco impaginato di ID di tutte le flotte a cui appartiene il veicolo. Gli esempi seguenti utilizzano AWS CLI.

Per recuperare un elenco di ID in pagine degli ID di tutte le flotte a cui il veicolo appartiene, eseguire il comando riportato di seguito comando riportato di seguito comando riportato di seguito.

Sostituisci *il nome del veicolo* con l'ID del veicolo.

```
aws iotfleetwise list-fleets-for-vehicle \  
  --vehicle-name vehicle-name
```

Puoi utilizzare l'operazione [ListVehiclesInFleet](#) API per recuperare un elenco impaginato di riepiloghi di tutti i veicoli di una flotta. Gli esempi seguenti utilizzano AWS CLI.

Per recuperare un elenco di riepiloghi in pagine di riepiloghi di tutti i veicoli di una flotta, eseguire il comando riportato di seguito comando riportato di seguito.

Sostituisci l'*ID della flotta* con l'ID della flotta.

```
aws iotfleetwise list-vehicles-in-fleet \  
  --fleet-id fleet-id
```

È possibile utilizzare l'operazione [GetFleet](#) API per recuperare le informazioni sulla flotta. Gli esempi seguenti utilizzano AWS CLI.

Per recuperare i metadati di un parco istanze, eseguire il comando riportato di seguito.

Sostituisci l'*ID della flotta* con l'ID della flotta.

```
aws iotfleetwise get-fleet \  
  --fleet-id fleet-id
```

Note

Questa operazione è [consistente finale](#). In altre parole, le modifiche alla flotta potrebbero non riflettersi immediatamente.

Raccogli e trasferisci dati con le campagne

Una campagna è un'orchestrazione delle regole di raccolta dei dati. Le campagne forniscono al FleetWise software Edge Agent for AWS IoT istruzioni su come selezionare, raccogliere e trasferire dati sul cloud.

Crei campagne nel cloud. Dopo che tu o il tuo team avete approvato una campagna, l'AWS IoT la distribuisce FleetWise automaticamente sui veicoli. Puoi scegliere di distribuire una campagna su un veicolo o su una flotta di veicoli. Il software Edge Agent non inizia a raccogliere dati finché non viene distribuita una campagna in corso sul veicolo.

Note

Le campagne non funzioneranno finché non avrai quanto segue.

- Il software Edge Agent è in esecuzione nel veicolo. Per ulteriori informazioni su come sviluppare, installare e utilizzare il software Edge Agent, procedi come segue.
 1. Passa alla [FleetWiseconsole AWS IoT](#).
 2. Nella home page del servizio, nella FleetWise sezione Guida introduttiva all'AWS IoT, scegli Explore Edge Agent.
- Hai predisposto il AWS IoT Core rifornimento del tuo veicolo. Per ulteriori informazioni, consulta [Veicoli di approvvigionamento](#).

Ogni campagna contiene le seguenti informazioni.

`signalCatalogArn`

L'Amazon Resource Name (ARN) del catalogo dei segnali associato alla campagna.

(Facoltativo) `tags`

I tag sono metadati che possono essere utilizzati per gestire la campagna. Puoi assegnare lo stesso tag a risorse di servizi diversi per indicare che le risorse sono correlate.

`TargetArn`

L'ARN di un veicolo o di una flotta su cui viene implementata la campagna.

name

Un nome univoco che aiuta a identificare la campagna.

collectionScheme

Gli schemi di raccolta dei dati forniscono al software Edge Agent istruzioni su quali dati raccogliere o quando raccogliarli. AWS L'IoT FleetWise attualmente supporta lo schema di raccolta basato sulle condizioni e lo schema di raccolta basato sul tempo.

conditionBasedCollectionScheme

Lo schema di raccolta basato sulle condizioni utilizza un'espressione logica per riconoscere quali dati raccogliere. Il software Edge Agent raccoglie i dati quando viene soddisfatta la condizione.

expression

L'espressione logica utilizzata per riconoscere quali dati raccogliere. Ad esempio, se viene specificata l'espressione ``myVehicle.InVehicleTemperature` > 50.0`, il software Edge Agent raccoglie valori di temperatura superiori a 50,0. Per istruzioni su come scrivere espressioni, vedere [Espressioni logiche per le campagne](#)

(Facoltativo) `triggerMode` può essere uno dei seguenti valori.

- **RISING_EDGE**— Il software Edge Agent raccoglie i dati solo quando la condizione viene soddisfatta per la prima volta. Ad esempio, ``myVehicle.AirBagDeployed` == true`.
- **ALWAYS**— Il software Edge Agent raccoglie dati ogni volta che viene soddisfatta la condizione.

(Facoltativo) `minimumTriggerIntervalMs`

La durata minima tra due eventi di raccolta dati, in millisecondi. Se un segnale cambia spesso, è possibile raccogliere dati a una velocità inferiore.

(Facoltativo) `conditionLanguageVersion`

La versione del linguaggio di espressione condizionale.

timeBasedCollectionScheme

Quando definisci uno schema di raccolta basato sul tempo, specifica un periodo di tempo in millisecondi. Il software Edge Agent utilizza il periodo di tempo per decidere con quale frequenza raccogliere i dati. Ad esempio, se il periodo di tempo è 120.000 millisecondi, il software Edge Agent raccoglie i dati una volta ogni due minuti.

(Facoltativo) `compression`

Per risparmiare larghezza di banda wireless e ridurre il traffico di rete, puoi specificare [SNAPPY](#) per comprimere i dati nei veicoli.

Per impostazione predefinita (OFF), il software Edge Agent non comprime i dati.

`dataDestinationConfigs`

Scegli la destinazione in cui la campagna trasferirà i dati del veicolo. Puoi scegliere di archiviare i dati in Amazon S3 o Amazon Timestream.

S3 è un meccanismo di archiviazione dei dati conveniente che offre funzionalità di gestione dei dati durevoli e servizi dati a valle. Puoi utilizzare S3 per i dati relativi ai comportamenti di guida o all'analisi della manutenzione a lungo termine.

Timestream è un meccanismo di persistenza dei dati che può aiutarti a identificare tendenze e modelli quasi in tempo reale. Puoi utilizzare Timestream per dati di serie temporali, ad esempio per analizzare le tendenze storiche della velocità o della frenata dei veicoli.

(Facoltativo) `dataExtraDimensions`

È possibile aggiungere uno o più attributi per fornire informazioni aggiuntive per un segnale.

(Facoltativo) `description`

Puoi aggiungere una descrizione per aiutare a identificare lo scopo della campagna.

(Facoltativo) `diagnosticsMode`

Quando la modalità diagnostica è configurata su `SEND_ACTIVE_DTCS`, la campagna invia codici diagnostici di errore (DTC) standard memorizzati che aiutano a identificare i problemi del veicolo. Ad esempio, P0097 indica che il modulo di controllo del motore (ECM) ha stabilito che l'ingresso del sensore di temperatura dell'aria aspirata 2 (IAT2) è inferiore al normale intervallo di sensori.

Per impostazione predefinita (OFF), il software Edge Agent non invia codici diagnostici.

(Facoltativo) `expiryTime`

Puoi definire la data di scadenza della tua campagna. Quando la campagna scade, il software Edge Agent interrompe la raccolta dei dati come specificato in questa campagna. Se sul veicolo vengono distribuite più campagne, il software Edge Agent utilizza altre campagne per raccogliere dati.

Valore predefinito: 253402243200 (31 dicembre 9999, 00:00:00 UTC)

(Facoltativo) `postTriggerCollectionDuration`

È possibile definire una durata della raccolta post-attivazione, in modo che il software Edge Agent continui a raccogliere dati per un periodo specificato dopo la chiamata di uno schema. Ad esempio, se viene richiamato uno schema di raccolta basato sulle condizioni con la seguente espressione: `$variable.`myVehicle.Engine.RPM` > 7000.0`, il software Edge Agent continua a raccogliere i valori di giri al minuto (RPM) per il motore. Anche se l'RPM supera i 7000 solo una volta, potrebbe indicare che c'è un problema meccanico. In questo caso, potresti volere che il software Edge Agent continui a raccogliere dati per monitorare la condizione.

Valore predefinito: 0

(Facoltativo) `priority`

Puoi specificare un numero intero per indicare il livello di priorità della campagna. Le campagne con un numero inferiore hanno priorità più elevate. Se distribuisce più campagne su un veicolo, le campagne con priorità più elevate vengono avviate per prime.

Valore predefinito: 0

(Facoltativo) `signalsToCollect`

Un elenco di segnali da cui vengono raccolti i dati quando viene richiamato lo schema di raccolta dei dati.

 Important

I segnali utilizzati nell'espressione per lo schema di raccolta basato sulle condizioni devono essere specificati in questo campo.

`name`

Il nome del segnale da cui vengono raccolti i dati quando viene richiamato lo schema di raccolta dei dati.

(Facoltativo) `maxSampleCount`

Il numero massimo di campioni di dati che il software Edge Agent raccoglie e trasferisce nel cloud quando viene richiamato lo schema di raccolta dei dati.

(Facoltativo) `minimumSamplingIntervalMs`

La durata minima del tempo tra due eventi di raccolta dei campioni di dati, in millisecondi. Se un segnale cambia spesso, puoi utilizzare questo parametro per raccogliere dati a una velocità inferiore.

Intervallo valido: 0-4294967295

(Facoltativo) `spoolingMode`

Se `spoolingMode` è configurato per `T0_DISK`, il software Edge Agent archivia temporaneamente i dati localmente quando un veicolo non è connesso al cloud. Dopo aver ristabilito la connessione, i dati archiviati localmente vengono trasferiti automaticamente nel cloud.

Valore predefinito: OFF

(Facoltativo) `startTime`

Una campagna approvata viene attivata all'inizio.

Valore predefinito: 0

Lo stato di una campagna può corrispondere a uno dei seguenti valori.

- **CREATING**— FleetWise L'AWSIoT sta elaborando la tua richiesta per creare la campagna.
- **WAITING_FOR_APPROVAL**— Una volta creata, una campagna entra nello **WAITING_FOR_APPROVAL** stato. Per approvare la campagna, utilizza l'operazione **UpdateCampaign** API. Dopo l'approvazione della campagna, l'AWSIoT la distribuisce FleetWise automaticamente sul veicolo o sulla flotta bersaglio. Per ulteriori informazioni, consulta [Aggiorna una campagna \(AWS CLI\)](#).
- **RUNNING** — La campagna è attiva.
- **SUSPENDED**— La campagna è sospesa. Per riprendere la campagna, utilizza l'operazione **UpdateCampaign** API.

AWSL' IoT FleetWise fornisce le seguenti operazioni API che puoi utilizzare per creare e gestire campagne.

- [CreateCampaign](#)— Crea una nuova campagna.
- [UpdateCampaign](#)— Aggiorna una campagna esistente. Dopo aver creato una campagna, è necessario utilizzare questa operazione API per approvare la campagna.

- [DeleteCampaign](#)— Elimina una campagna esistente.
- [ListCampaigns](#)— Recupera un elenco impaginato di riepiloghi per tutte le campagne.
- [GetCampaign](#)— Recupera informazioni su una campagna.

Tutorial

- [Creazione di una campagna](#)
- [Aggiorna una campagna \(AWS CLI\)](#)
- [Eliminare una campagna](#)
- [Ottieni informazioni sulla campagna \(\) AWS CLI](#)

Creazione di una campagna

Puoi utilizzare la FleetWise console o l'API AWS IoT per creare campagne per raccogliere i dati dei veicoli.

Important

Affinché la tua campagna funzioni, devi disporre di quanto segue:

- Il software Edge Agent è in esecuzione nel veicolo. Per ulteriori informazioni su come sviluppare, installare e utilizzare il software Edge Agent, procedi come segue:
 1. Passa alla [FleetWiseconsole AWS IoT](#).
 2. Nella home page del servizio, nella FleetWise sezione Guida introduttiva all'AWSIoT, scegli Explore Edge Agent.
- Hai predisposto il AWS IoT Core rifornimento del tuo veicolo. Per ulteriori informazioni, consulta [Veicoli di approvvigionamento](#).

Argomenti

- [Crea una campagna \(console\)](#)
- [Crea una campagna \(\) AWS CLI](#)
- [Espressioni logiche per le campagne](#)

Crea una campagna (console)

Puoi utilizzare la FleetWise console AWS IoT per creare una campagna per selezionare, raccogliere e trasferire i dati dei veicoli sul cloud.

Per creare una campagna

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. Nel riquadro di navigazione, scegli Campagne.
3. Nella pagina Campagne, scegli Crea campagna, quindi completa i passaggi indicati nei seguenti argomenti.

Argomenti

- [Fase 1: Configurare la campagna](#)
- [Fase 2: Definire la destinazione di archiviazione](#)
- [Fase 3: Aggiungi veicoli](#)
- [Fase 4: Revisione e creazione](#)
- [Passaggio 5: Implementa una campagna](#)

Important

- È necessario disporre di un catalogo di segnali e di un veicolo prima di creare una campagna. Per ulteriori informazioni, consultare [Crea e gestisci cataloghi di segnali](#) e [Crea, fornisci e gestisci veicoli](#).
- Dopo aver creato una campagna, devi approvarla. Per ulteriori informazioni, consulta [Passaggio 5: Implementa una campagna](#).

Fase 1: Configurare la campagna

In Informazioni generali, procedi come segue:

1. Inserisci un nome per la campagna.
2. (Opzionale) Immettere una descrizione.

Configura lo schema di raccolta dati della campagna. Uno schema di raccolta dati fornisce al software Edge Agent istruzioni su quali dati raccogliere o quando raccoglierli. Nella FleetWise console AWS IoT, puoi configurare uno schema di raccolta dati nei seguenti modi:

- Definire manualmente lo schema di raccolta dei dati.
- Carica un file per definire automaticamente lo schema di raccolta dei dati.

Nell'opzione di configurazione, scegliete una delle seguenti opzioni:

- Per specificare manualmente il tipo di schema di raccolta dati e definire le opzioni per personalizzare lo schema, scegliete Definisci schema di raccolta dati.

Specificate manualmente il tipo di schema di raccolta dati e definite le opzioni per personalizzare lo schema.

1. Nella sezione Dettagli dello schema di raccolta dati, scegli il tipo di schema di raccolta dati che desideri venga utilizzato da questa campagna. Per utilizzare un'espressione logica per riconoscere i dati del veicolo da raccogliere, scegli Basato sulle condizioni. Per utilizzare un periodo di tempo specifico per decidere con quale frequenza raccogliere i dati del veicolo, scegli Basato sul tempo.
2. Definisci la durata della raccolta dei dati da parte della campagna.

Note

Per impostazione predefinita, una campagna approvata viene attivata immediatamente e non ha un'ora di fine impostata. Per evitare costi aggiuntivi, devi specificare un intervallo di tempo.

3. Se è stato specificato uno schema di raccolta dati basato sulle condizioni, è necessario definire un'espressione logica per riconoscere i dati da raccogliere. AWS IoT FleetWise utilizza un'espressione logica per riconoscere quali dati raccogliere per uno schema basato sulle condizioni. L'espressione deve specificare il nome completo di un segnale come variabile, operatore di confronto e valore di confronto.

Ad esempio, se si specifica l'espressione ``myVehicle.InVehicleTemperature` > 50.0`, AWS IoT FleetWise raccoglie valori di temperatura superiori a 50,0. Per istruzioni su come scrivere espressioni, consulta [Espressioni logiche per le campagne](#)

Immettere l'espressione logica utilizzata per riconoscere i dati da raccogliere.

4. (Facoltativo) È possibile specificare la versione linguistica dell'espressione condizionale. Il valore predefinito è 1.
 5. (Facoltativo) È possibile specificare l'intervallo minimo di attivazione, che è il periodo di tempo minimo tra due eventi di raccolta dati. Ad esempio, se un segnale cambia spesso, potresti voler raccogliere i dati a una velocità inferiore.
 6. Specificare la condizione della modalità Trigger per la raccolta dei dati da parte del software Edge Agent. Per impostazione predefinita, il FleetWise software Edge Agent for AWS IoT raccoglie sempre i dati ogni volta che viene soddisfatta la condizione. In alternativa, può raccogliere dati solo quando la condizione viene soddisfatta per la prima volta, al primo trigger.
 7. Se è stato specificato uno schema di raccolta dati basato sul tempo, è necessario specificare un periodo di tempo, in millisecondi, compreso tra 10.000 e 60.000 millisecondi. Il software Edge Agent utilizza il periodo di tempo per decidere con quale frequenza raccogliere i dati.
 8. (Facoltativo) È possibile modificare le opzioni dello schema avanzato dello schema.
 - a. Per risparmiare larghezza di banda wireless e ridurre il traffico di rete comprimendo i dati, scegli Snappy.
 - b. (Facoltativo) Per definire per quanto tempo, in millisecondi, continuare a raccogliere dati dopo un evento di raccolta dati, puoi specificare la durata della raccolta Post Trigger.
 - c. (Facoltativo) Per indicare il livello di priorità della campagna, puoi specificare la priorità della campagna. Le campagne con un numero di priorità inferiore vengono implementate per prime e sono considerate con una priorità più elevata.
 - d. Il software Edge Agent può archiviare temporaneamente i dati localmente quando un veicolo non è connesso al cloud. Dopo aver ristabilito la connessione, i dati archiviati localmente vengono trasferiti automaticamente nel cloud. Specificate se desiderate che Edge Agent memorizzi i dati localmente durante una connessione interrotta.
 - e. (Facoltativo) Per fornire informazioni aggiuntive su un segnale, aggiungi fino a cinque attributi come dimensioni dati aggiuntive.
- Per caricare un file per definire lo schema di raccolta dei dati, seleziona Carica un file.json dal tuo dispositivo locale. AWS L'IoT definisce FleetWise automaticamente le opzioni che è possibile definire nel file. È possibile rivedere e aggiornare le opzioni selezionate.

Carica un file.json con dettagli sullo schema di raccolta dei dati.

1. Per importare informazioni sullo schema di raccolta dati, scegli Scegli file. Per ulteriori informazioni sul formato di file richiesto, consulta la documentazione dell'[CreateCampaignAPI](#).

 Note

AWS FleetWise Attualmente IoT supporta l'estensione del formato di file.json.

2. AWS IoT definisce FleetWise automaticamente lo schema di raccolta dei dati in base alle informazioni contenute nel file. Esamina le opzioni che l'AWS IoT ha FleetWise selezionato per te. Puoi aggiornare le opzioni, se necessario.

Specificare i segnali

È possibile specificare i segnali da cui raccogliere i dati quando viene richiamato lo schema di raccolta dati.

 Important

I segnali utilizzati nell'espressione per lo schema di raccolta basato sulle condizioni devono essere specificati in questo campo.

Per specificare i segnali da cui raccogliere i dati

1. Cerca il nome completo del segnale.

 Note

Il nome completo del segnale è il percorso del segnale più il nome del segnale. Utilizzate un punto (.) per fare riferimento a un segnale secondario.

Ad esempio,

`Vehicle.Chassis.SteeringWheel.HandsOff.HandsOffSteeringState`

è il nome completo dell'`HandsOffSteeringState` attuatore.

`Vehicle.Chassis.SteeringWheel.HandsOff.` è il percorso verso questo attuatore.

2. (Facoltativo) Per Numero massimo di campioni, inserite il numero massimo di campioni di dati che il software Edge Agent raccoglie e trasferisce nel cloud quando viene richiamato lo schema di raccolta dei dati.

3. (Facoltativo) Per Intervallo di campionamento minimo, immettete la durata minima del tempo tra due eventi di raccolta dei campioni di dati, in millisecondi. Se un segnale cambia spesso, puoi utilizzare questo parametro per raccogliere dati a una velocità inferiore.
4. Per aggiungere un altro segnale, scegli Aggiungi altri segnali. Puoi aggiungere fino a 999 segnali.
5. Seleziona Avanti.

Fase 2: Definire la destinazione di archiviazione

Note

Puoi trasferire i dati del veicolo ad Amazon S3 solo se la campagna contiene segnali di dati del sistema di visione.

I dati del sistema di visione sono disponibili in anteprima e sono soggetti a modifiche.

Scegli la destinazione in cui desideri archiviare i dati raccolti dalla campagna. Puoi trasferire i dati del veicolo su Amazon S3 o Amazon Timestream.

Nelle impostazioni di destinazione, procedi come segue:

- Scegli S3 o Timestream dall'elenco a discesa.

Per archiviare i dati del veicolo in un bucket S3, scegli Amazon S3. S3 è un servizio di storage di oggetti che archivia i dati come oggetti all'interno di bucket. Per ulteriori informazioni, consulta [Creazione, configurazione e utilizzo dei bucket Amazon S3](#) nella Guida per l'utente di Amazon Simple Storage Service.

S3 ottimizza il costo dell'archiviazione dei dati e fornisce meccanismi aggiuntivi per utilizzare i dati dei veicoli, come data lake, archiviazione centralizzata dei dati, pipeline di elaborazione dei dati e analisi. Puoi utilizzare S3 per archiviare i dati per l'elaborazione e l'analisi in batch. Ad esempio, puoi creare report di eventi di frenata brusca per il tuo modello di machine learning (ML). I dati del veicolo in entrata vengono memorizzati nel buffer per 10 minuti prima della consegna.

Amazon S3

Important

Puoi trasferire dati su S3 solo se AWS IoT FleetWise dispone delle autorizzazioni per scrivere nel bucket S3. Per ulteriori informazioni sulla concessione dell'accesso, consulta [Controllare l'accesso con AWS IoT FleetWise](#).

Nelle impostazioni di destinazione S3, procedi come segue:

1. Per il bucket S3, scegli un bucket con le autorizzazioni per AWS IoT FleetWise
2. (Facoltativo) Inserisci un prefisso personalizzato che puoi utilizzare per organizzare i dati archiviati nel bucket S3.
3. Scegli il formato di output, ovvero il formato dei file che vengono salvati come nel bucket S3.
4. Scegli se vuoi comprimere i dati memorizzati nel bucket S3 come file.gzip. Consigliamo di comprimere i dati perché riduce al minimo i costi di archiviazione.
5. Le opzioni selezionate nelle impostazioni di destinazione S3 modificano l'URI dell'oggetto S3 di esempio. Questo è un esempio di come i file vengono salvati in S3.

Per archiviare i dati del veicolo in una tabella Timestream, scegli Amazon Timestream. Puoi utilizzare Timestream per interrogare i dati dei veicoli in modo da identificare tendenze e modelli. Ad esempio, puoi utilizzare Timestream per creare un allarme per il livello di carburante del veicolo. I dati del veicolo in arrivo vengono trasferiti su Timestream quasi in tempo reale. Per ulteriori informazioni, consulta [Cos'è Amazon Timestream?](#) nella Amazon Timestream Developer Guide.

Amazon Timestream

Important

Puoi trasferire dati su una tabella solo se l'AWSIoT FleetWise dispone delle autorizzazioni per scrivere dati in Timestream. Per ulteriori informazioni sulla concessione dell'accesso, consulta [Controllare l'accesso con AWS IoT FleetWise](#).

Nelle impostazioni della tabella Timestream, procedi come segue:

1. Per il nome del database Timestream, scegli il nome del tuo database Timestream dall'elenco a discesa.
2. Per il nome della tabella Timestream, scegli il nome della tabella Timestream dall'elenco a discesa.

In Accesso al servizio per Timestream, procedi come segue:

- Scegli un ruolo IAM dall'elenco a discesa.
- Seleziona Avanti.

Fase 3: Aggiungi veicoli

Per scegliere su quali veicoli distribuire la tua campagna, selezionali nell'elenco dei veicoli. Filtra i veicoli cercando gli attributi e i relativi valori aggiunti durante la creazione dei veicoli o in base al nome del veicolo.

In Filtra veicoli, procedi come segue:

1. Nella casella di ricerca, trova l'attributo o il nome del veicolo e selezionalo dall'elenco.

Note

Ogni attributo può essere utilizzato una sola volta.

2. Inserisci il valore dell'attributo o il nome del veicolo su cui desideri distribuire la campagna. Ad esempio, se il nome completo dell'attributo è `fuelType`, inseriscilo `gasoline` come valore.
3. Per cercare un altro attributo del veicolo, ripeti i passaggi precedenti. Puoi cercare fino a cinque attributi del veicolo e un numero illimitato di nomi di veicoli.
4. I veicoli che corrispondono alla tua ricerca sono elencati sotto Nome veicolo. Scegli i veicoli su cui vuoi che venga lanciata la campagna.

Note

Nei risultati di ricerca vengono visualizzati fino a 100 veicoli. Scegli Seleziona tutto per aggiungere tutti i veicoli alla campagna.

5. Seleziona Avanti.

Fase 4: Revisione e creazione

Verifica le configurazioni per la campagna, quindi scegli Crea campagna.

Note

Dopo aver creato una campagna, tu o il tuo team dovete distribuirla sui veicoli.

Passaggio 5: Implementa una campagna

Dopo aver creato una campagna, tu o il tuo team dovete distribuirla sui veicoli.

Per distribuire una campagna

1. Nella pagina di riepilogo della campagna, scegli Distribuisci.
2. Verifica e conferma che desideri avviare l'implementazione e iniziare a raccogliere dati dai veicoli collegati alla campagna.
3. Seleziona Deploy (Implementa).

Se desideri sospendere la raccolta dei dati dai veicoli collegati alla campagna, nella pagina di riepilogo della campagna, scegli Sospendi. Per riprendere la raccolta di dati dai veicoli collegati alla campagna, scegli Riprendi.

Crea una campagna () AWS CLI

Puoi utilizzare l'operazione [CreateCampaign](#) API per creare una campagna. Gli esempi seguenti utilizzano AWS CLI.

Quando crei una campagna, i dati raccolti dai veicoli possono essere archiviati in Amazon S3 (S3) o Amazon Timestream. Scegli Timestream per un database di serie temporali veloce, scalabile e senza server, ad esempio per archiviare dati che richiedono un'elaborazione quasi in tempo reale. Scegli S3 per uno storage di oggetti con scalabilità, disponibilità dei dati, sicurezza e prestazioni leader del settore.

⚠ Important

Puoi trasferire i dati del veicolo solo se l'AWS IoT FleetWise dispone delle autorizzazioni per scrivere dati in S3 o Timestream. Per ulteriori informazioni sulla concessione dell'accesso, consulta [Controllare l'accesso con AWS IoT FleetWise](#).

Crea una campagna

⚠ Important

- È necessario disporre di un catalogo di segnali e di un veicolo o di una flotta prima di creare una campagna. Per ulteriori informazioni, consulta [Crea e gestisci cataloghi di segnali](#), [Crea, fornisci e gestisci veicoli](#) e [Crea e gestisci flotte](#).
- Dopo aver creato una campagna, devi utilizzare l'operazione UpdateCampaign API per approvarla. Per ulteriori informazioni, consultare [Aggiorna una campagna \(AWS CLI\)](#)

Per creare una campagna, esegui il comando seguente.

Sostituisci *file-name* con il nome del file JSON che contiene la configurazione della campagna.

```
aws iotfleetwise create-campaign --cli-input-json file://file-name.json
```

- Sostituisci *il nome della campagna* con il nome della campagna che stai creando.
- Sostituisci *signal-catalog-arn* con l'Amazon Resource Name (ARN) del catalogo dei segnali.
- Sostituisci *target-arn* con l'ARN di una flotta o di un veicolo che hai creato.
- Sostituisci *bucket-arn* con l'ARN del bucket S3.

```
{  
  "name": "campaign-name",  
  "targetArn": "target-arn",  
  "signalCatalogArn": "signal-catalog-arn",  
  "collectionScheme": {  
    "conditionBasedCollectionScheme": {
```

```

        "conditionLanguageVersion": 1,
        "expression": "$variable.`Vehicle.DemoBrakePedalPressure` > 7000",
        "minimumTriggerIntervalMs": 1000,
        "triggerMode": "ALWAYS"
    }
},
"compression": "SNAPPY",
"diagnosticsMode": "OFF",
"postTriggerCollectionDuration": 1000,
"priority": 0,
"signalsToCollect": [
    {
        "maxSampleCount": 100,
        "minimumSamplingIntervalMs": 0,
        "name": "Vehicle.DemoEngineTorque"
    },
    {
        "maxSampleCount": 100,
        "minimumSamplingIntervalMs": 0,
        "name": "Vehicle.DemoBrakePedalPressure"
    }
],
"spoolingMode": "TO_DISK",
"dataDestinationConfigs": [
    {
        "s3Config": {
            "bucketArn": "bucket-arn",
            "dataFormat": "PARQUET",
            "prefix": "campaign-name",
            "storageCompressionFormat": "GZIP"
        }
    }
]
}

```

- Sostituisci il *nome della campagna* con il nome della campagna che stai creando.
- Sostituisci *signal-catalog-arn* con l'Amazon Resource Name (ARN) del catalogo dei segnali.
- Sostituisci *target-arn* con l'ARN di una flotta o di un veicolo che hai creato.
- Sostituisci *role-arn* con l'ARN del ruolo di esecuzione dell'attività che concede l' FleetWise autorizzazione AWS IoT a fornire dati alla tabella Timestream.
- Sostituisci *table-arn* con l'ARN della tabella Timestream.

```
{
  "name": "campaign-name",
  "targetArn": "target-arn",
  "signalCatalogArn": "signal-catalog-arn",
  "collectionScheme": {
    "conditionBasedCollectionScheme": {
      "conditionLanguageVersion": 1,
      "expression": "$variable.`Vehicle.DemoBrakePedalPressure` > 7000",
      "minimumTriggerIntervalMs": 1000,
      "triggerMode": "ALWAYS"
    }
  },
  "compression": "SNAPPY",
  "diagnosticsMode": "OFF",
  "postTriggerCollectionDuration": 1000,
  "priority": 0,
  "signalsToCollect": [
    {
      "maxSampleCount": 100,
      "minimumSamplingIntervalMs": 0,
      "name": "Vehicle.DemoEngineTorque"
    },
    {
      "maxSampleCount": 100,
      "minimumSamplingIntervalMs": 0,
      "name": "Vehicle.DemoBrakePedalPressure"
    }
  ],
  "spoolingMode": "TO_DISK",
  "dataDestinationConfigs": [
    {
      "timestreamConfig": {
        "executionRoleArn": "role-arn",
        "timestreamTableArn": "table-arn"
      }
    }
  ]
}
```

Espressioni logiche per le campagne

AWS IoT FleetWise utilizza un'espressione logica per riconoscere quali dati raccogliere come parte di una campagna. Per ulteriori informazioni sulle espressioni, consulta [Expressions](#) nella AWS IoT Events Developer Guide.

La variabile expression deve essere costruita in modo da rispettare le regole relative al tipo di dati raccolti. Per i dati del sistema di telemetria, la variabile di espressione deve essere il nome completo del segnale. Per i dati del sistema di visione, l'espressione combina il nome completo del segnale con il percorso che conduce dal tipo di dati del segnale a una delle sue proprietà.

Ad esempio, se il catalogo dei segnali contiene i seguenti nodi:

```
{
  myVehicle.ADAS.Camera:
    type: sensor
    datatype: Vehicle.ADAS.CameraStruct
    description: "A camera sensor"

  myVehicle.ADAS.CameraStruct:
    type: struct
    description: "An obstacle detection camera output struct"
}
```

Se i nodi seguono la definizione di ROS 2:

```
{
  Vehicle.ADAS.CameraStruct.msg:
    boolean obstaclesExists
    uint8[] image
    Obstacle[30] obstacles
}
{
  Vehicle.ADAS.Obstacle.msg:
    float32: probability
    uint8 o_type
    float32: distance
}
```

Le seguenti sono tutte le possibili variabili di espressione degli eventi:

```
{
```

```

...
    $variable.`myVehicle.ADAS.Camera.obstaclesExists`
    $variable.`myVehicle.ADAS.Camera.Obstacle[0].probability`
    $variable.`myVehicle.ADAS.Camera.Obstacle[1].probability`
...
    $variable.`myVehicle.ADAS.Camera.Obstacle[29].probability`
    $variable.`myVehicle.ADAS.Camera.Obstacle[0].o_type`
    $variable.`myVehicle.ADAS.Camera.Obstacle[1].o_type`
...
    $variable.`myVehicle.ADAS.Camera.Obstacle[29].o_type`
    $variable.`myVehicle.ADAS.Camera.Obstacle[0].distance`
    $variable.`myVehicle.ADAS.Camera.Obstacle[1].distance`
...
    $variable.`myVehicle.ADAS.Camera.Obstacle[29].distance`
}

```

Aggiorna una campagna (AWS CLI)

Puoi utilizzare l'operazione [UpdateCampaign](#) API per aggiornare una campagna esistente. Il comando seguente utilizza AWS CLI.

- Sostituisci *campaign-name* con il nome della campagna che stai aggiornando.
- Sostituisci *l'azione* con una delle seguenti:
 - APPROVE— Approva la campagna per consentire all'AWS IoT FleetWise di implementarlo su un veicolo o una flotta.
 - SUSPEND— Sospende la campagna. La campagna viene eliminata dai veicoli e tutti i veicoli della campagna sospesa interromperanno l'invio di dati.
 - RESUME— Riattiva la campagna. SUSPEND La campagna viene ridistribuita su tutti i veicoli e i veicoli riprenderanno a inviare dati.
 - UPDATE— Aggiorna la campagna definendo gli attributi e associandoli a un segnale.

```

aws iotfleetwise update-campaign \
    --name campaign-name \
    --action action

```

Eliminare una campagna

Puoi utilizzare la FleetWise console o l'API AWS IoT per eliminare le campagne.

Eliminare una campagna (console)

Per eliminare una campagna, usa la FleetWise console AWS IoT.

Per eliminare una campagna

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. Nel riquadro di navigazione, scegli Campagne.
3. Nella pagina Campagne, scegli la campagna di destinazione.
4. Scegli Elimina.
5. In Elimina**campaign-name?** , inserisci il nome della campagna da eliminare, quindi scegli Conferma.

Eliminare una campagna (AWS CLI)

Puoi utilizzare l'operazione [DeleteCampaign](#)API per eliminare una campagna. Nell'esempio seguente viene utilizzato AWS CLI.

Per eliminare una campagna, esegui il comando seguente.

Sostituisci *il nome della campagna* con il nome del veicolo che stai eliminando.

```
aws iotfleetwise delete-campaign --name campaign-name
```

Ottieni informazioni sulla campagna () AWS CLI

Puoi utilizzare l'operazione [ListCampaigns](#)API per verificare se una campagna è stata eliminata. Gli esempi seguenti utilizzano AWS CLI.

Per recuperare un elenco impaginato di riepiloghi per tutte le campagne, esegui il comando seguente.

```
aws iotfleetwise list-campaigns
```

Puoi utilizzare l'operazione [GetCampaign](#) API per recuperare le informazioni sul veicolo. Gli esempi seguenti utilizzano AWS CLI.

Per recuperare i metadati di una campagna, esegui il comando seguente.

Sostituisci *campaign-name* con il nome della campagna che desideri recuperare.

```
aws iotfleetwise get-campaign --name campaign-name
```

Note

Questa operazione è [consistente finale](#). In altre parole, le modifiche alla campagna potrebbero non essere applicate immediatamente.

Elaborazione e visualizzazione dei dati del veicolo

Il FleetWise software Edge Agent for AWS IoT trasferisce i dati selezionati del veicolo su Amazon Timestream o Amazon Simple Storage Service (Amazon S3). Una volta che i dati arrivano nella destinazione dei dati, puoi utilizzare altri AWS servizi per visualizzarli e condividerli.

Elaborazione dei dati del veicolo in Timestream

Timestream è un database di serie temporali completamente gestito in grado di archiviare e analizzare trilioni di punti dati di serie temporali al giorno. I tuoi dati vengono archiviati in una tabella Timestream gestita dal cliente. Puoi utilizzare Timestream per interrogare i dati dei veicoli in modo da ottenere informazioni dettagliate sui veicoli. Per ulteriori informazioni, consulta [Cos'è Amazon Timestream?](#)

Lo schema predefinito dei dati trasferiti su Timestream contiene i seguenti campi.

Nome del campo	Tipo di dati	Description
eventId	varchar	L'ID dell'evento di raccolta dati.
vehicleName	varchar	L'ID del veicolo da cui sono stati raccolti i dati.
name	varchar	Il nome della campagna utilizzato dal software Edge Agent per raccogliere dati.
time	timestamp	Il timestamp del punto dati.
measure_name	varchar	Il nome del segnale.
measure_value::bigint	bigint	Valori del segnale di tipo Integer.

Nome del campo	Tipo di dati	Description
measure_v alue::double	double	Valori dei segnali di tipo Double.
measure_v alue::boolean	booleano	Valori dei segnali di tipo booleano.

Visualizzazione dei dati del veicolo memorizzati in Timestream

Dopo aver trasferito i dati del veicolo su Timestream, puoi utilizzare i seguenti AWS servizi per visualizzare, monitorare, analizzare e condividere i tuoi dati.

- Visualizza e monitora i dati nelle dashboard utilizzando Grafana [o Amazon Managed Grafana](#). Puoi visualizzare i dati da più AWS fonti (come Amazon CloudWatch e Timestream) e altre fonti di dati con un'unica dashboard Grafana.
- [Analizza e visualizza i dati nelle dashboard utilizzando Amazon. QuickSight](#)

Elaborazione dei dati del veicolo in S3

Amazon S3 è un servizio di storage di oggetti che archivia e protegge qualsiasi quantità di dati. Puoi usare S3 per una varietà di casi d'uso, come data lake, backup e ripristino, archiviazione, applicazioni aziendali, AWS IoT dispositivi e analisi dei big data. I tuoi dati vengono archiviati in S3 come oggetti in bucket. Per ulteriori informazioni, consulta [Cos'è Amazon S3?](#)

Lo schema predefinito dei dati trasferiti su Amazon S3 contiene i seguenti campi.

Nome del campo	Tipo di dati	Description
eventId	varchar	L'ID dell'evento di raccolta dati.
vehicleName	varchar	L'ID del veicolo da cui sono stati raccolti i dati.

Nome del campo	Tipo di dati	Description
name	varchar	Il nome della campagna utilizzato dal software Edge Agent per raccogliere dati.
time	timestamp	Il timestamp del punto dati.
measure_name	varchar	Il nome del segnale.
measure_value_BIGINT	bigint	Valori del segnale di tipo Integer.
measure_value_DOUBLE	double	Valori dei segnali di tipo Double.
measure_value_BOOLEAN	booleano	Valori dei segnali di tipo booleano.
measure_value_STRUCT	struct	Valori dei segnali di tipo Struct.

Formato oggetto S3

AWS IoT FleetWise trasferisce i dati del veicolo su S3 dove vengono salvati come oggetto. Puoi utilizzare l'URI dell'oggetto che identifica in modo univoco i dati per trovare i dati della campagna. Il formato URI degli oggetti S3 dipende dal fatto che i dati raccolti siano dati non strutturati o elaborati.

Dati non strutturati

I dati non strutturati vengono archiviati in S3 in modo non predefinito. Può essere in vari formati, come immagini o video.

I messaggi del veicolo passati all'AWS IoT FleetWise con dati di segnale provenienti dai file Amazon Ion vengono decodificati e trasferiti su S3 come oggetti. Gli oggetti S3 rappresentano ogni segnale e sono codificati in modo binario.

L'URI dell'oggetto S3 di dati non strutturati utilizza il seguente formato:

```
s3://bucket-name/prefix/unstructured-data/random-ID-yyyy-MM-dd-HH-mm-ss-SSS-vehicleName-signalName-fieldName
```

Dati elaborati

I dati elaborati vengono archiviati in S3 e sottoposti a fasi di elaborazione che convalidano, arricchiscono e trasformano i messaggi. Gli elenchi di oggetti e la velocità sono esempi di dati elaborati.

I dati trasferiti su S3 vengono archiviati come oggetti che rappresentano record memorizzati nel buffer per un periodo di circa 10 minuti. Per impostazione predefinita, AWS IoT FleetWise aggiunge un prefisso orario UTC nel formato `year=YYYY/month=MM/date=DD/hour=HH` prima di scrivere oggetti su S3. Questo prefisso crea una gerarchia logica nel bucket in cui ogni barra in avanti (/) crea un livello nella gerarchia. I dati elaborati contengono anche l'URI dell'oggetto S3 relativo ai dati non strutturati.

L'URI dell'oggetto S3 dei dati elaborati utilizza il seguente formato:

```
s3://bucket-name/prefix/processed-data/year=YYYY/month=MM/day=DD/hour=HH/part-0000-random-ID.gz.parquet
```

Dati non elaborati

I dati grezzi, noti anche come dati primari, sono dati raccolti dai file Amazon Ion. Puoi utilizzare i dati grezzi per risolvere eventuali problemi o per causare errori alla radice.

L'URI dell'oggetto S3 con dati grezzi utilizza il seguente formato:

```
s3://bucket-name/prefix/raw-data/vehicle-name/eventID-timestamp.10n
```

Analisi dei dati del veicolo memorizzati in S3

Dopo aver trasferito i dati del veicolo su S3, puoi utilizzare i seguenti AWS servizi per monitorare, analizzare e condividere i tuoi dati.

Estrai e analizza i dati utilizzando Amazon SageMaker per i flussi di lavoro downstream di etichettatura e machine learning (ML).

Per ulteriori informazioni, consulta i seguenti argomenti nell'Amazon SageMaker Developer Guide:

- [Dati di processo](#)
- [Addestra modelli di machine learning](#)
- [Immagini delle etichette](#)

Cataloga i tuoi dati utilizzando Crawler di AWS Glue e analizzali in Amazon Athena. Per impostazione predefinita, gli oggetti scritti su S3 hanno partizioni temporali in stile Apache Hive, con percorsi di dati che contengono coppie chiave-valore collegate da segni uguali.

Per ulteriori informazioni, consulta i seguenti argomenti nella Guida per l'utente di Amazon Athena:

- [Partizionamento dei dati in Athena](#)
- [Utilizzo AWS Glue per connettersi a sorgenti dati in Amazon S3](#)
- [Procedure consigliate per l'utilizzo di Athena con AWS Glue](#)

Visualizza i dati utilizzando Amazon QuickSight leggendo direttamente la tabella Athena o il bucket S3.

 Tip

Se stai leggendo direttamente da S3, conferma che i dati del veicolo siano in formato JSON perché Amazon QuickSight non supporta il formato Apache Parquet.

Per ulteriori informazioni, consulta i seguenti argomenti nella Amazon QuickSight User Guide:

- [Fonti di dati supportate](#)
- [Creazione di una fonte di dati](#)

AWS CLI e AWSSDK

In questa sezione vengono fornite informazioni sulla creazione di richieste API AWS IoT FleetWise. Per ulteriori informazioni su AWS IoT FleetWise [operazioni e tipi di dati](#), consulta [AWS IoT FleetWise Riferimento alle API](#).

Da utilizzare AWS IoT FleetWise con una varietà di linguaggi di programmazione, usa il [AWSSDK](#), che contengono le seguenti funzionalità automatiche:

- Firma crittografica delle richieste di servizio
- Nuovi tentativi di richiesta
- Gestione delle risposte di errore

Per accedere alla riga di comando, utilizzare AWS IoT FleetWise con il [AWS CLI](#). Puoi controllare AWS IoT FleetWise e gli altri servizi, dalla riga di comando, automatizzandoli tramite script.

Risoluzione dei problemi AWS IoT FleetWise

Utilizza le informazioni e le soluzioni per la risoluzione dei problemi in questa sezione per risolvere i problemi relativi all'AWS IoT FleetWise.

Le seguenti informazioni possono aiutarti a risolvere i problemi più comuni relativi all'AWS IoT FleetWise.

Argomenti

- [Problemi relativi al manifesto del decoder](#)
- [Problemi relativi al FleetWise software Edge Agent per AWS IoT](#)

Problemi relativi al manifesto del decoder

Risolvi i problemi relativi al manifesto del decoder.

Diagnosi delle chiamate all'API manifesto del decoder

Errore	Linee guida per la risoluzione dei problemi
<code>UpdateOperationFailure.ConflictingDecoderUpdate</code>	Lo stesso manifesto del decoder presenta più richieste di aggiornamento. Attendi e riprova.
<code>UpdateOperationFailure.InternalFailure</code>	<code>InternalFailure</code> viene lanciato come eccezione incapsulata. Il problema stesso dipende dall'eccezione incapsulata.
<code>UpdateOperationFailure.ActiveDecoderUpdate</code>	Il manifesto del decodificatore è in uno <code>Active</code> stato e non può essere aggiornato. Cambia lo stato del manifesto del decoder in <code>DRAFT</code> , quindi riprova.
<code>UpdateOperationFailure.ConflictingModelUpdate</code>	AWS FleetWise L'iot sta cercando di eseguire la convalida rispetto a un modello di veicolo (model manifest) modificato da qualcun altro. Attendi e riprova.

Errore	Linee guida per la risoluzione dei problemi
<pre>UpdateOperationFailure.Mode ManifestValidationResponse : FailureReason.MODEL_DATA_ENTRIES_NOT_FOUND</pre>	<p>Al modello di veicolo non è associato alcun segnale. Aggiungi segnali al modello del veicolo e verifica che i segnali siano disponibili nel catalogo dei segnali associato.</p>
<pre>UpdateOperationFailure.Mode ManifestValidationResponse : FailureReason.MODEL_NOT_ACTIVE</pre>	<p>Aggiorna il modello del veicolo in modo che sia ACTIVE a posto, quindi riprova.</p>
<pre>UpdateOperationFailure.Mode ManifestValidationResponse : FailureReason.MODEL_NOT_FOUND</pre>	<p>AWSL'IoT non FleetWise riesce a trovare il modello di veicolo associato al manifesto del decoder. Verifica l'Amazon Resource Name (ARN) del modello di veicolo e riprova.</p>
<pre>UpdateOperationFailure.Mode ManifestValidationResponse (FailureReason.MODEL_DATA_ENTRIES_READ_FAILURE)</pre>	<p>La convalida del modello di veicolo non è riuscita perché i nomi dei segnali del modello di veicolo non sono stati trovati nel catalogo dei segnali. Verifica che i segnali del modello di veicolo siano tutti inclusi nel catalogo dei segnali associato.</p>
<pre>UpdateOperationFailure.ValidationFailure</pre>	<p>Nella richiesta di aggiornamento del manifesto del decoder sono stati trovati segnali o interfacce e di rete non validi. Verifica che tutti i segnali e le interfacce di rete restituiti dall'eccezione esistano, che tutti i segnali utilizzati siano associati a un'interfaccia disponibile e che non rimuoverai un'interfaccia a cui sono associati segnali.</p>
<pre>UpdateOperationFailure.KmsKeyAccessDenied</pre>	<p>C'è un problema di autorizzazione sulla chiave AWS Key Management Service (AWS KMS) utilizzata per l'operazione. Verifica di utilizzare un ruolo con accesso alla chiave e riprova.</p>

Errore	Linee guida per la risoluzione dei problemi
<code>UpdateOperationFailure.DecoderDoesNotExist</code>	Il manifesto del decodificatore non esiste. Verifica il nome del manifesto del decoder e riprova.

I messaggi di errore relativi ai dati del sistema di visione con il `SIGNAL_DECODER_INCOMPATIBLE_WITH_SIGNAL_CATALOG` motivo includeranno un suggerimento nella risposta che fornisce informazioni sul motivo per cui la richiesta non è riuscita. È possibile utilizzare il suggerimento per determinare quali linee guida per la risoluzione dei problemi seguire.

Note

I dati del sistema di visione sono in versione di anteprima e sono soggetti a modifiche.

Diagnosi, decodificatore, manifesto, convalida dei dati del sistema di visione

Errore	Linee guida per la risoluzione dei problemi
<code>InvalidSignalDecoder.withReason(SignalDecoderFailureReason.NO_SIGNAL_IN_CATALOG_FOR_DECODER_SIGNAL)</code>	AWSL'IoT FleetWise non ha trovato la struttura del segnale principale utilizzata nel decodificatore di segnale utilizzando il catalogo dei segnali. Verificate che il segnale principale della struttura sia definito correttamente nel catalogo dei segnali.
<code>InvalidSignalDecoder.withReason(SignalDecoderFailureReason.SIGNAL_DECODER_TYPE_INCOMPATIBLE_WITH_MESSAGE_SIGNAL_TYPE)</code>	Un messaggio primitivo nel catalogo dei segnali non è stato definito con lo stesso tipo di dati nella richiesta di aggiornamento del manifesto del decoder. Verificate che i messaggi primitivi definiti nella richiesta corrispondano alla definizione del catalogo dei segnali corrispondente.

Errore	Linee guida per la risoluzione dei problemi
<pre>InvalidSignalDecoder.withReason(SignalDecoderFailureReason.STRUCT_SIZE_MISMATCH)</pre>	<p>Il numero di proprietà definite in una struttura nel catalogo dei segnali non corrisponde al numero di proprietà che state tentando di decodificare nel manifesto del decoder. Verifica di avere il numero corretto di segnali da decodificare confrontandolo con i segnali definiti nel catalogo dei segnali.</p>
<pre>InvalidSignalDecoder.withReason(SignalDecoderFailureReason.SIGNAL_DECODER_INCOMPATIBLE_WITH_SIGNAL_CATALOG)</pre>	<p>AWSIoT FleetWise ha trovato un segnale definito come STRUCT nel catalogo dei segnali senza una structuredMessageDefinition definizione nella richiesta del manifesto del decoder. Assicurati che ogni struttura sia definita come richiesta di structuredMessageDefinition aggiornamento del manifesto del decoder.</p>
<pre>InvalidSignalDecoder.withReason(SignalDecoderFailureReason.SIGNAL_DECODER_INCOMPATIBLE_WITH_SIGNAL_CATALOG)</pre>	<p>Il segnale principale della struttura utilizzata nel manifesto del decoder non è definito correttamente come struttura nel catalogo dei segnali. La struttura del segnale principale utilizzata nel manifesto del decoder deve avere il relativo campo structFullyQualified Name definito. Inoltre, necessita di un nodo STRUCT.fullyQualifiedName</p>
<pre>InvalidSignalDecoder.withReason(SignalDecoderFailureReason.SIGNAL_DECODER_INCOMPATIBLE_WITH_SIGNAL_CATALOG)</pre>	<p>Uno dei messaggi leaf utilizzati nella richiesta del manifesto del decoder non è definito come messaggio primitivo. Verificate che tutti gli oggetti leaf nella richiesta siano definiti come messaggi primitivi.</p>

Errore	Linee guida per la risoluzione dei problemi
<pre>InvalidSignalDecoder.withReason(SignalDecoderFailureReason.SIGNAL_DECODER_INCOMPATIBLE_WITH_SIGNAL_CATALOG)</pre>	<p>Un oggetto array nel catalogo dei segnali non è stato definito come <code>structuredMessageList</code> definizione nella richiesta di aggiornamento del manifesto del decoder. Verificate che tutte le proprietà dell'array siano definite come <code>structuredMessageList Definition</code> nella richiesta di aggiornamento del manifesto del decoder.</p>

Problemi relativi al FleetWise software Edge Agent per AWS IoT

Risolvi i problemi relativi al software Edge Agent.

Problemi

- [Problema: il software Edge Agent non si avvia.](#)
- [Problema: \[ERRORE\] \[IoTFleetWiseEngine: :connect\]: \[Impossibile avviare la libreria di persistenza\]](#)
- [Problema: il software Edge Agent non raccoglie i PID di diagnostica di bordo \(OBD\) II e i codici di errore diagnostici \(DTC\).](#)
- [Problema: il FleetWise software Edge Agent for AWS IoT non raccoglie dati dalla rete o non è in grado di applicare regole di ispezione dei dati.](#)
- [Problema: \[ERROR\] \[AwsIotConnectivityModule: :connect\]: \[Connessione fallita con errore\] o \[WARN\] \[AwsIotChannel: :send\]: \[Nessuna connessione MQTT attiva.\]](#)

Problema: il software Edge Agent non si avvia.

È possibile che vengano visualizzati i seguenti errori quando il software Edge Agent non si avvia.

- ```
Error from reader: * Line 1, Column 1
Syntax error: value, object or array expected.
```

Soluzione: assicurati che il file di configurazione del FleetWise software Edge Agent for AWS IoT utilizzi un formato JSON valido. Ad esempio, verifica che le virgole siano utilizzate correttamente.

Per ulteriori informazioni sul file di configurazione, procedi come segue per scaricare la Guida per gli sviluppatori del FleetWise software Edge Agent for AWS IoT.

1. Passa alla [FleetWiseconsole AWS IoT](#).
2. Nella home page del servizio, nella FleetWise sezione Guida introduttiva all'AWSIoT, scegli Explore Edge Agent.

```
[ERROR] [SocketCANBusChannel::connect]: [SocketCan with name xxx is not accessible]
[ERROR] [IoTFleetWiseEngine::connect]: [Failed to Bind Consumers to Producers]
```

Soluzione: questo errore potrebbe verificarsi quando il software Edge Agent non riesce a stabilire una comunicazione socket con le interfacce di rete definite nel file di configurazione.

Per verificare che tutte le interfacce di rete definite nella configurazione siano disponibili, esegui il comando seguente.

```
ip link show
```

Per portare online un'interfaccia di rete, esegui il comando seguente. Sostituisci *network-interface-id* con l'ID dell'interfaccia di rete.

```
sudo ip link set network-interface-id up
```

```
[ERROR] [AwsIotConnectivityModule::connect]: [Connection failed with error]
[WARN] [AwsIotChannel::send]: [No alive MQTT Connection.]
or
[WARN] [AwsIotChannel::send]: [aws-c-common: AWS_ERROR_FILE_INVALID_PATH]
```

Soluzione: questo errore potrebbe verificarsi quando il software Edge Agent non riesce a stabilire una connessione MQTT aAWS IoT Core. Verifica che quanto segue sia configurato correttamente e riavvia il software Edge Agent.

- `mqttConnection::endpointUrl`— endpoint del dispositivo IoT dell'AWSaccount.
- `mqttConnection::clientId`— L'ID del veicolo su cui è in esecuzione il software Edge Agent.
- `mqttConnection::certificateFilename`— Il percorso del file del certificato del veicolo.
- `mqttConnection::privateKeyFilename`— Il percorso del file della chiave privata del veicolo.

- L'hai utilizzato AWS IoT Core per rifornire il veicolo. Per ulteriori informazioni, consulta [Veicoli di approvvigionamento](#).

Per ulteriori informazioni sulla risoluzione dei problemi, consulta [SDK per dispositivi AWS IoT per C++Domande frequenti](#).

## Problema: [ERRORE] [IoTFleetWiseEngine: :connect]: [Impossibile avviare la libreria di persistenza]

Soluzione: potresti visualizzare questo errore quando il software Edge Agent non riesce a localizzare l'archiviazione di persistenza. Verifica che quanto segue sia configurato correttamente e riavvia il software Edge Agent.

`persistency:persistencyPath`— Un percorso locale utilizzato per mantenere gli schemi di raccolta, i manifesti del decodificatore e le istantanee dei dati.

## Problema: il software Edge Agent non raccoglie i PID di diagnostica di bordo (OBD) II e i codici di errore diagnostici (DTC).

Soluzione: potresti visualizzare questo errore se `obdInterface:pidRequestIntervalSeconds` o è configurato su 0. `obdInterface:dtcRequestIntervalSeconds`

Se il software Edge Agent è in esecuzione in un veicolo con trasmissione automatica, assicurati che `obdInterface:hasTransmissionEcu` sia configurato su `true`.

Se il veicolo supporta gli ID di arbitrato CAN bus (Extended Controller Area Network), assicurati che `obdInterface:useExtendedIds` sia configurato su `true`.

## Problema: il FleetWise software Edge Agent for AWS IoT non raccoglie dati dalla rete o non è in grado di applicare regole di ispezione dei dati.

Soluzione: potresti visualizzare questo errore quando vengono violate le quote predefinite.

| Risorsa                    | Quota                                                    | Regolabile | Nota                                                          |
|----------------------------|----------------------------------------------------------|------------|---------------------------------------------------------------|
| Valore dell'ID del segnale | L'ID del segnale deve essere inferiore o uguale a 50.000 | Sì         | Il software Edge Agent non raccoglierà dati da segnali con un |

| Risorsa                                              | Quota | Regolabile | Nota                                                                                                                                            |
|------------------------------------------------------|-------|------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                      |       |            | ID superiore a 50.000.<br>Ti consigliamo di controllare quanti segnali contiene il catalogo dei segnali prima di modificare questa quota.       |
| Numero di schemi di raccolta dati attivi per veicolo | 256   | Sì         | Ti consigliamo di controllare quante campagne hai creato nel cloud e quanti schemi contiene ciascuna campagna prima di modificare questa quota. |
| Dimensione del buffer di cronologia dei segnali      | 20 MB | Sì         | Se la quota viene violata, il software Edge Agent interrompe e la raccolta di nuovi dati.                                                       |

**Problema:** [ERROR] [AwsIotConnectivityModule: :connect]: [Connessione fallita con errore] o [WARN] [AwsIotChannel: :send]: [Nessuna connessione MQTT attiva.]

**Soluzione:** potresti visualizzare questo errore quando il software Edge Agent non è connesso al cloud. Per impostazione predefinita, il software Edge Agent invia una richiesta di ping a AWS IoT Core ogni minuto e attende tre minuti. Se non c'è risposta, il software Edge Agent ristabilisce automaticamente la connessione al cloud.

# Sicurezza nell' AWS IoT FleetWise

La sicurezza del cloud AWS è la massima priorità. In qualità di AWS cliente, puoi beneficiare di un data center e di 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 modello 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. I revisori esterni testano e verificano regolarmente l'efficacia della nostra sicurezza nell'ambito dei [AWS Programmi di AWS conformità dei Programmi di conformità](#) dei di . Per ulteriori informazioni sui programmi di conformità che si applicano all' AWS IoT FleetWise, consulta [AWS Services in Scope by Compliance Program](#) .
- Sicurezza nel cloud: la tua responsabilità è determinata dal AWS servizio che utilizzi. Sei anche responsabile di altri fattori, tra cui la riservatezza dei dati, i tuoi requisiti aziendali e le leggi e le normative applicabili

Questa documentazione aiuta a capire come applicare il modello di responsabilità condivisa quando si utilizza l' AWS IoT FleetWise. Ti mostra come configurare l' AWS IoT FleetWise per soddisfare i tuoi obiettivi di sicurezza e conformità. Imparerai anche a utilizzare altri AWS servizi che ti aiutano a monitorare e proteggere le tue FleetWise risorse AWS IoT.

## Indice

- [Protezione dei dati nell' AWS IoT FleetWise](#)
- [Controllo dell'accesso con AWS IoT FleetWise](#)
- [Identity and Access Management per AWS IoT FleetWise](#)
- [Convalida della conformità per l'IoT AWS FleetWise](#)
- [Resilienza nell'IoT AWS FleetWise](#)
- [Sicurezza dell'infrastruttura nell' AWS IoT FleetWise](#)
- [Analisi della configurazione e delle vulnerabilità nell'IoT AWS FleetWise](#)
- [Best practice di sicurezza per l' AWS IoT FleetWise](#)

# Protezione dei dati nell' AWS IoT FleetWise

Il modello di [responsabilità AWS condivisa modello](#) di si applica alla protezione dei dati nell' AWS IoT FleetWise. Come descritto in questo modello, AWS è responsabile della protezione dell'infrastruttura globale che gestisce tutto il Cloud AWS. L'utente è responsabile del controllo dei contenuti ospitati su questa infrastruttura. L'utente è inoltre responsabile della configurazione della protezione e delle attività di gestione per i Servizi AWS utilizzati. Per ulteriori informazioni sulla privacy dei dati, vedi le [Domande frequenti sulla privacy dei dati](#). Per informazioni sulla protezione dei dati in Europa, consulta il post del blog relativo al [Modello di responsabilità condivisa AWS e GDPR](#) nel Blog sulla sicurezza AWS .

Ai fini della protezione dei dati, consigliamo di proteggere Account AWS le credenziali e configurare i singoli utenti con AWS IAM Identity Center or AWS Identity and Access Management (IAM). In tal modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere i suoi compiti. Ti suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- Usa SSL/TLS per comunicare con le risorse. AWS È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Configura l'API e la registrazione delle attività degli utenti con. AWS CloudTrail
- Utilizza soluzioni di AWS crittografia, insieme a tutti i controlli di sicurezza predefiniti all'interno Servizi AWS.
- Utilizza i servizi di sicurezza gestiti avanzati, come Amazon Macie, che aiutano a individuare e proteggere i dati sensibili archiviati in Amazon S3.
- Se hai bisogno di moduli crittografici convalidati FIPS 140-2 per l'accesso AWS tramite un'interfaccia a riga di comando o un'API, utilizza un endpoint FIPS. Per ulteriori informazioni sugli endpoint FIPS disponibili, consulta il [Federal Information Processing Standard \(FIPS\) 140-2](#).

Ti consigliamo vivamente di non inserire mai informazioni riservate o sensibili, ad esempio gli indirizzi e-mail dei clienti, nei tag o nei campi di testo in formato libero, ad esempio nel campo Nome. Ciò include quando lavori con l' AWS IoT FleetWise o altro Servizi AWS utilizzando la console, l'API o AWS gli SDK. AWS CLI I dati inseriti nei tag o nei campi di testo in formato libero utilizzati per i nomi possono essere utilizzati per i la fatturazione o i log di diagnostica. Quando fornisci un URL a un server esterno, ti suggeriamo vivamente di non includere informazioni sulle credenziali nell'URL per convalidare la tua richiesta al server.

AWS FleetWise L'IoT è destinato all'uso con un Edge Agent sviluppato e installato sull'hardware del veicolo supportato per trasmettere i dati del veicolo al AWS Cloud. L'estrazione di dati dai veicoli potrebbe essere soggetta alle normative sulla privacy dei dati in alcune giurisdizioni. Prima di utilizzare l' AWS IoT FleetWise e installare Edge Agent, ti consigliamo vivamente di valutare i tuoi obblighi di conformità ai sensi della legge applicabile. Ciò include tutti i requisiti legali applicabili per fornire avvisi sulla privacy legalmente adeguati e ottenere i consensi necessari per l'estrazione dei dati del veicolo.

## Crittografia a riposo

I dati raccolti da un veicolo vengono trasmessi al cloud tramite un AWS IoT Core messaggio con il protocollo di messaggi MQTT. AWS L'IoT FleetWise fornisce i dati al tuo database Amazon Timestream. In Timestream, i tuoi dati sono crittografati. Per impostazione predefinita, tutti Servizi AWS crittografano i dati inattivi.

Encryption at rest si integra con AWS Key Management Service (AWS KMS) per gestire la chiave di crittografia utilizzata per crittografare i dati. Puoi scegliere di utilizzare una chiave gestita dal cliente per crittografare i dati raccolti dall' AWS IoT FleetWise. Puoi creare, gestire e visualizzare la tua chiave di crittografia tramite AWS KMS. Per ulteriori informazioni, consulta [Cos'è AWS Key Management Service?](#) nella Guida per gli AWS Key Management Service sviluppatori.

## Crittografia in transito

Tutti i dati scambiati con AWS IoT i servizi vengono crittografati in transito utilizzando Transport Layer Security (TLS). Per ulteriori informazioni, consulta l'argomento relativo alla [sicurezza del trasporto](#) nella Guida per gli sviluppatori AWS IoT .

Inoltre, AWS IoT Core supporta [l'autenticazione e l'autorizzazione](#) per aiutare a controllare in modo sicuro l'accesso alle FleetWise risorse AWS IoT. I veicoli possono utilizzare i certificati X.509 per autenticarsi (accedere) all'uso dell' AWS IoT FleetWise e utilizzare AWS IoT Core le policy per ottenere l'autorizzazione (disporre delle autorizzazioni) per eseguire azioni specifiche. Per ulteriori informazioni, consulta [the section called “Veicoli di approvvigionamento”](#).

## Crittografia dei dati

La crittografia dei dati si riferisce alla protezione dei dati durante il transito (mentre viaggiano da e verso l' AWS IoT FleetWise e tra gateway e server) e quando sono inattivi (mentre sono archiviati su dispositivi locali o all'interno). Servizi AWS È possibile proteggere i dati inattivi utilizzando la crittografia lato client.

### Note

AWS L'elaborazione FleetWise edge IoT espone le API ospitate all'interno di FleetWise gateway AWS IoT e accessibili tramite la rete locale. Queste API sono esposte su una connessione TLS supportata da un certificato server di proprietà del connettore IoT AWS Edge. FleetWise Per l'autenticazione del client, queste API utilizzano una password di controllo degli accessi. La chiave privata del certificato del server e la password di controllo degli accessi sono entrambe archiviate su disco. AWS L'elaborazione FleetWise edge IoT si basa sulla crittografia del file system per la sicurezza di queste credenziali archiviate.

Per ulteriori informazioni sulla crittografia lato server e sulla crittografia lato client, consulta gli argomenti elencati di seguito.

### Indice

- [Crittografia a riposo](#)
- [Gestione delle chiavi](#)

## Crittografia a riposo

AWS L'loT FleetWise archivia i tuoi dati nel AWS cloud e sui gateway.

### Dati inattivi nel cloud AWS

AWS L'loT FleetWise archivia i dati in un altro Servizi AWS ambiente che, per impostazione predefinita, crittografa i dati inattivi. Encryption at rest si integra con [AWS Key Management Service \(AWS KMS\)](#) per la gestione della chiave di crittografia utilizzata per crittografare i valori delle proprietà degli asset e i valori aggregati in IoT. AWS FleetWise Puoi scegliere di utilizzare una chiave gestita dal cliente per crittografare i valori delle proprietà degli asset e aggregare i valori in IoT AWS . FleetWise Puoi creare, gestire e visualizzare la tua chiave di crittografia tramite. AWS KMS

Puoi scegliere una chiave Chiave di proprietà di AWS o una chiave gestita dal cliente per crittografare i tuoi dati.

### Come funziona

Encryption at rest si integra con AWS KMS la gestione della chiave di crittografia utilizzata per crittografare i dati.

- Chiave di proprietà di AWS — Chiave di crittografia predefinita. AWS L'IoT FleetWise possiede questa chiave. Non puoi visualizzare, gestire o utilizzare questa chiave nel tuo Account AWS. Inoltre, non puoi visualizzare le operazioni sulla chiave nei AWS CloudTrail registri. È possibile utilizzare questa chiave senza costi aggiuntivi.
- Chiave gestita dal cliente: la chiave viene memorizzata nel tuo account, che crei, possiedi e gestisci. Hai il pieno controllo sulla chiave KMS. AWS KMS Si applicano costi aggiuntivi.

## Chiavi di proprietà di AWS

Chiavi di proprietà di AWS non sono archiviate nel tuo account. Fanno parte di una raccolta di chiavi KMS che AWS possiede e gestisce per essere utilizzate in più Account AWS lingue. Servizi AWS può essere utilizzato Chiavi di proprietà di AWS per proteggere i tuoi dati.

Non puoi visualizzarne, gestirli Chiavi di proprietà di AWS, utilizzarli o controllarne l'utilizzo. Tuttavia, non è necessario intraprendere alcuna azione o modificare alcun programma per proteggere le chiavi che crittografano i dati.

Se le utilizzi Chiavi di proprietà di AWS non ti verrà addebitata alcuna commissione e non vengono conteggiate nelle AWS KMS quote del tuo account.

## Chiavi gestite dal cliente

Le chiavi gestite dal cliente sono chiavi KMS nel tuo account create da te, di tua proprietà e gestite da te. Hai il pieno controllo su queste chiavi KMS, come le seguenti:

- Stabilire e mantenere le proprie politiche chiave, le politiche IAM e le sovvenzioni
- Abilitarli e disabilitarli
- Ruotando il loro materiale crittografico
- Aggiungere tag
- Creazione di alias che si riferiscono ad essi
- Pianificazione della loro eliminazione

Puoi anche utilizzare CloudTrail Amazon CloudWatch Logs per tenere traccia delle richieste a cui l'AWS IoT FleetWise invia per tuo AWS KMS conto.

Se utilizzi chiavi gestite dai clienti, devi concedere FleetWise l'accesso AWS IoT alla chiave KMS memorizzata nel tuo account. AWS L'IoT FleetWise utilizza la crittografia a busta e la gerarchia delle

chiavi per crittografare i dati. La chiave di crittografia di AWS KMS viene utilizzata per criptare la chiave root di questa gerarchia. Per ulteriori informazioni, consulta [Crittografia envelope](#) nella Guida per gli sviluppatori di AWS Key Management Service .

La seguente policy di esempio concede FleetWise le autorizzazioni AWS IoT per creare una chiave gestita dal cliente per tuo conto.

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "Stmt1603902045292",
 "Action": [
 "kms:GenerateDataKey*",
 "kms:Decrypt",
 "kms:DescribeKey",
 "kms:CreateGrant",
 "kms:RetireGrant",
 "kms:RevokeGrant"
],
 "Effect": "Allow",
 "Resource": "*"
 }
]
}
```

### Important

Quando aggiungi le nuove sezioni alla tua politica sulle chiavi KMS, non modificare le sezioni esistenti nella politica. AWS L' IoT non FleetWise può eseguire operazioni sui dati se la crittografia è abilitata per l' AWS IoT FleetWise e si verifica una delle seguenti condizioni:

- La chiave KMS è disabilitata o eliminata.
- La politica delle chiavi KMS non è configurata correttamente per il servizio.

## Utilizzo dei dati del sistema di visione con crittografia a riposo

### Note

I dati del sistema di visione sono in versione di anteprima e sono soggetti a modifiche.

Se hai una crittografia gestita dal cliente con AWS KMS chiavi abilitate sul tuo FleetWise account AWS IoT e desideri utilizzare i dati del sistema di visione, ripristina le impostazioni di crittografia per renderle compatibili con tipi di dati complessi. Ciò consente FleetWise all' AWS IoT di stabilire autorizzazioni aggiuntive necessarie per i dati del sistema di visione.

### Note

Il manifesto del decodificatore potrebbe essere bloccato in uno stato di convalida se non hai ripristinato le impostazioni di crittografia per i dati del sistema di visione.

1. Utilizza l'operazione [GetEncryptionConfiguration](#) API per verificare se la AWS KMS crittografia è abilitata. Non sono necessarie ulteriori azioni se il tipo di crittografia è `FLEETWISE_DEFAULT_ENCRYPTION`.
2. Se il tipo di crittografia è `KMS_BASED_ENCRYPTION`, utilizza l'operazione [PutEncryptionConfiguration](#) API per reimpostare il tipo di crittografia su `FLEETWISE_DEFAULT_ENCRYPTION`.

```
{
 aws iotfleetwise put-encryption-configuration --encryption-type
 FLEETWISE_DEFAULT_ENCRYPTION
}
```

3. Utilizza l'operazione [PutEncryptionConfiguration](#) API per riattivare il tipo di crittografia su `KMS_BASED_ENCRYPTION`.

```
{
 aws iotfleetwise put-encryption-configuration \
 --encryption-type "KMS_BASED_ENCRYPTION"
 --kms-key-id kms_key_id
}
```

Per ulteriori informazioni sull'attivazione della crittografia, vedere [Gestione delle chiavi](#).

## Gestione delle chiavi

### AWS Gestione delle chiavi FleetWise cloud IoT

Per impostazione predefinita, AWS IoT FleetWise utilizza Chiavi gestite da AWS per proteggere i dati in Cloud AWS. Puoi aggiornare le impostazioni per utilizzare una chiave gestita dal cliente per crittografare i dati in AWS IoT FleetWise. Puoi creare, gestire e visualizzare la tua chiave di crittografia tramite AWS Key Management Service (AWS KMS).

AWS L'IoT FleetWise supporta la crittografia lato server con chiavi gestite dal cliente archiviate AWS KMS per crittografare i dati per le seguenti risorse.

| AWS FleetWise<br>Risorsa IoT               | Tipo di dati | Campi crittografati inattivi con chiavi gestite dal cliente |
|--------------------------------------------|--------------|-------------------------------------------------------------|
| Catalogo dei segnali                       |              | description                                                 |
|                                            | Attributo    | descrizione, AllowedValues, defaultValue, min, max          |
|                                            | Actuator     | descrizione, AllowedValues, min, max                        |
|                                            | Sensor       | descrizione, AllowedValues, min, max                        |
| Modello di veicolo (manifesto del modello) |              | description                                                 |
| Manifesto del decoder                      |              | description                                                 |
|                                            | CanInterface | Nome del protocollo, versione del protocollo                |
|                                            | ObdInterface | requestMessageId, dtcRequestInterval Secondi,, Standard OBD |

|                              |                                |                                                                                                          |
|------------------------------|--------------------------------|----------------------------------------------------------------------------------------------------------|
| AWS FleetWise<br>Risorsa IoT | Tipo di dati                   | Campi crittografati inattivi con chiavi gestite dal cliente                                              |
|                              |                                | hasTransmissionEcu, Secondi, pidRequestInterval useExtendedIds                                           |
|                              | CanSignal                      | fattore, isSigned isBigEndian, lunghezza, messageId, offset, startBit                                    |
|                              | ObdSignal                      | ByteLength, offset, pid, scaling, ServiceMode, startByte,, pidResponseLength bitMaskLength bitRightShift |
| Veicolo                      |                                | attributes                                                                                               |
| Campagna                     |                                | description                                                                                              |
|                              | conditionBasedCollectionSchema | espressione,, minimumTriggerInterval Ms conditionLanguageVersion, TriggerMode                            |
|                              | TimeBasedCollectionScheme      | Periodo/MS                                                                                               |

#### Note

Altri dati e risorse vengono crittografati utilizzando la crittografia predefinita con chiavi gestite da AWS IoT FleetWise. Questa chiave viene creata e archiviata nell' FleetWise account AWS IoT.

Per ulteriori informazioni, consulta [Cos'è AWS Key Management Service?](#) nella Guida per gli AWS Key Management Service sviluppatori.

Abilita la crittografia utilizzando le chiavi KMS (console)

Per utilizzare le chiavi gestite dai clienti con AWS IoT FleetWise, devi aggiornare FleetWise le impostazioni AWS IoT.

## Per abilitare la crittografia utilizzando le chiavi KMS (console)

1. Apri la [FleetWise console AWS IoT](#).
2. Vai a Impostazioni.
3. In Crittografia, scegli Modifica per aprire la pagina Modifica crittografia.
4. Per Tipo di chiave di crittografia, scegli Scegli una AWS KMS chiave diversa. Ciò consente la crittografia con chiavi gestite dal cliente archiviate in AWS KMS.

### Note

Puoi utilizzare solo la crittografia a chiave gestita dal cliente per FleetWise le risorse AWS IoT. Ciò include il catalogo dei segnali, il modello del veicolo (manifesto del modello), il manifesto del decodificatore, il veicolo, la flotta e la campagna.

5. Scegli la tua chiave KMS con una delle seguenti opzioni:
  - Per utilizzare una chiave KMS esistente: scegli l'alias della tua chiave KMS dall'elenco.
  - Per creare una nuova chiave KMS, scegli Crea una chiave. AWS KMS

### Note

Questo apre la AWS KMS console. Per ulteriori informazioni sulla creazione di una chiave KMS, consulta [Creating keys](#) nella AWS Key Management Service Developer Guide.

6. Scegli Salva per aggiornare le impostazioni.

## Abilita la crittografia utilizzando le chiavi KMS (AWS CLI)

Puoi utilizzare l'operazione [PutEncryptionConfiguration](#) API per abilitare la crittografia per il tuo FleetWise account AWS IoT. L'esempio seguente utilizza AWS CLI.

Per abilitare la crittografia, esegui il comando seguente.

- Sostituisci l'*id della chiave KMS* con l'ID della chiave KMS.

```
aws iotfleetwise put-encryption-configuration --kms-key-id KMS key id --encryption-type
KMS_BASED_ENCRYPTION
```

## Example response

```
{
 "kmsKeyId": "customer_kms_key_id",
 "encryptionStatus": "PENDING",
 "encryptionType": "KMS_BASED_ENCRYPTION"
}
```

## Policy della chiave KMS

Dopo aver creato una chiave KMS, devi almeno aggiungere la seguente dichiarazione alla tua politica delle chiavi KMS affinché funzioni con l'IoT AWS . FleetWise

```
{
 "Sid": "Allow FleetWise to encrypt and decrypt data when customer managed KMS key
based encryption is enabled",
 "Effect": "Allow",
 "Principal": {
 "Service": "iotfleetwise.amazonaws.com"
 },
 "Action": [
 "kms:GenerateDataKey*",
 "kms:Decrypt",
 "kms:DescribeKey",
 "kms:CreateGrant",
 "kms:RetireGrant",
 "kms:RevokeGrant"
],
 "Resource": "*"
}
```

Per ulteriori informazioni sulla modifica di una policy chiave KMS da utilizzare con AWS IoT FleetWise, consulta [Changing a key policy](#) nella AWS Key Management Service Developer Guide.

**⚠ Important**

Quando aggiungi le nuove sezioni alla tua politica delle chiavi KMS, non modificare le sezioni esistenti nella politica. AWS IoT FleetWise può eseguire operazioni sui dati se la crittografia è abilitata per l' AWS IoT FleetWise e si verifica una delle seguenti condizioni:

- La chiave KMS è disabilitata o eliminata.
- La politica delle chiavi KMS non è configurata correttamente per il servizio.

## Controllo dell'accesso con AWS IoT FleetWise

Le seguenti sezioni spiegano come controllare l'accesso da e verso le AWS IoT FleetWise risorse. Le informazioni fornite includono come concedere l'accesso alle applicazioni in modo che l' AWS IoT FleetWise possa trasferire i dati dei veicoli durante le campagne. Descrivono inoltre come concedere AWS IoT FleetWise l'accesso al bucket Amazon S3 (S3) o al database e alla tabella Amazon Timestream per archiviare i dati.

La tecnologia per gestire tutte queste forme di accesso è (IAM). AWS Identity and Access Management Per ulteriori informazioni su IAM, consulta [Che cos'è IAM?](#).

### Indice

- [Concedi AWS IoT FleetWise l'accesso a una destinazione Amazon S3](#)
- [Concedi AWS IoT FleetWise l'accesso a una destinazione Amazon Timestream](#)

## Concedi AWS IoT FleetWise l'accesso a una destinazione Amazon S3

Quando utilizzi una destinazione Amazon S3, AWS IoT FleetWise invia i dati del veicolo al tuo bucket S3 e, facoltativamente, puoi utilizzare una AWS KMS chiave di tua proprietà per la crittografia dei dati. Se la registrazione degli errori è abilitata, invia AWS IoT FleetWise anche gli errori di consegna dei dati al gruppo di log e ai CloudWatch flussi. È necessario disporre di un ruolo IAM durante la creazione di un flusso di distribuzione.

AWS IoT FleetWise utilizza una policy bucket con il responsabile del servizio per la destinazione S3. Per ulteriori informazioni sull'aggiunta di policy bucket, consulta [Aggiungere una policy bucket utilizzando la console Amazon S3 nella Amazon Simple Storage Service User Guide](#).

Utilizza la seguente politica di accesso per abilitare l'accesso al tuo AWS IoT FleetWise bucket S3. Se non sei proprietario del bucket S3, aggiungi `s3:PutObjectACL` all'elenco delle operazioni Amazon S3. Ciò garantisce al proprietario del bucket l'accesso completo agli oggetti forniti da AWS IoT FleetWise. Per ulteriori informazioni su come proteggere l'accesso agli oggetti nei bucket, consulta gli [esempi di policy Bucket](#) nella Amazon Simple Storage Service User Guide.

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Principal": {
 "Service": [
 "iotfleetwise.amazonaws.com"
]
 },
 "Action": [
 "s3:ListBucket"
],
 "Resource": "arn:aws:s3:::bucket-name"
 },
 {
 "Effect": "Allow",
 "Principal": {
 "Service": [
 "iotfleetwise.amazonaws.com"
]
 },
 "Action": [
 "s3:GetObject",
 "s3:PutObject"
],
 "Resource": "arn:aws:s3:::bucket-name/*",
 "Condition": {
 "StringEquals": {
 "aws:SourceArn": "campaign-arn",
 "aws:SourceAccount": "account-id"
 }
 }
 }
]
}
```

La seguente politica sui bucket si applica a tutte le campagne di un account in una AWS regione.

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Effect": "Allow",
 "Principal": {
 "Service": [
 "iotfleetwise.amazonaws.com"
]
 },
 "Action": [
 "s3:ListBucket"
],
 "Resource": "arn:aws:s3:::bucket-name"
 },
 {
 "Effect": "Allow",
 "Principal": {
 "Service": [
 "iotfleetwise.amazonaws.com"
]
 },
 "Action": [
 "s3:GetObject",
 "s3:PutObject"
],
 "Resource": "arn:aws:s3:::bucket-name/*",
 "Condition": {
 "StringLike": {
 "aws:SourceArn": "arn:aws:iotfleetwise:region:account-id:campaign/*",
 "aws:SourceAccount": "account-id"
 }
 }
 }
]
}
```

Se hai una chiave KMS collegata al tuo bucket S3, la chiave richiederà la seguente politica. Per informazioni sulla gestione delle chiavi, consulta [Protezione dei dati utilizzando la crittografia lato server con AWS Key Management Service chiavi \(SSE-KMS\) nella Guida per l'utente di Amazon Simple Storage Service](#).

```
{
 "Version": "2012-10-17",
 "Effect": "Allow",
 "Principal": {
 "Service": "iotfleetwise.amazonaws.com"
 },
 "Action": [
 "kms:GenerateDataKey",
 "kms:Decrypt"
],
 "Resource": "key-arn"
}
```

### Important

Quando crei un bucket, S3 crea una lista di controllo degli accessi (ACL) predefinita che garantisce al proprietario della risorsa il pieno controllo sulla risorsa. Se l'AWS IoT non è in FleetWise grado di fornire dati a S3, assicurati di disabilitare l'ACL sul bucket S3. Per ulteriori informazioni, consulta [Disabilitazione degli ACL per tutti i nuovi bucket e applicazione della proprietà degli oggetti nella Guida](#) per l'utente di Amazon Simple Storage Service.

## Concedi AWS IoT FleetWise l'accesso a una destinazione Amazon Timestream

Quando utilizzi una destinazione Timestream, AWS IoT FleetWise invia i dati del veicolo a una tabella Timestream. È necessario collegare le policy al ruolo IAM per consentire l'invio di dati AWS IoT FleetWise a Timestream.

Se utilizzi la console per [creare una campagna](#), l'AWS IoT FleetWise associa automaticamente la policy richiesta al ruolo.

Prima di iniziare, controlla quanto segue:

### Important

- È necessario utilizzare la stessa AWS regione quando si creano risorse Timestream per IoT AWS . FleetWise Se cambi AWS regione, potresti avere problemi ad accedere alle risorse Timestream.

- AWS FleetWise L'IoT è disponibile negli Stati Uniti orientali (Virginia settentrionale) e in Europa (Francoforte).
- Per l'elenco delle regioni supportate, consulta gli [endpoint e le quote di Timestream](#) in. Riferimenti generali di AWS

- È necessario disporre di un database Timestream. Per un tutorial, consulta [Creare un database](#) nella Amazon Timestream Developer Guide.
- Devi avere una tabella creata nel database Timestream specificato. Per un tutorial, consulta [Creare una tabella](#) nella Amazon Timestream Developer Guide.

Puoi usare il AWS CLI per creare un ruolo IAM con una policy di fiducia per Timestream. Per creare un ruolo IAM, esegui il comando seguente.

Per creare un ruolo IAM con una policy di fiducia

- Sostituiscilo *TimestreamExecutionRole* con il nome del ruolo che stai creando.
- Sostituisci *trust-policy* con il file JSON che contiene la policy di fiducia.

```
aws iam create-role --role-name TimestreamExecutionRole --assume-role-policy-document
file://trust-policy.json
```

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "timestreamTrustPolicy",
 "Effect": "Allow",
 "Principal": {
 "Service": "iotfleetwise.amazonaws.com"
 },
 "Action": "sts:AssumeRole",
 "Condition": {
 "StringEquals": {
 "aws:SourceArn": [
 "arn:aws:iotfleetwise:region:account-id:campaign/campaign-name"
]
 }
 }
 }
]
}
```

```

 "aws:SourceAccount": [
 "account-id"
]
 }
}
]
}

```

Crea una politica di autorizzazioni per concedere all' AWS IoT FleetWise le autorizzazioni per scrivere dati in Timestream. Per creare una politica di autorizzazioni, esegui il comando seguente.

Per creare una politica di autorizzazioni

- Sostituiscila *AWSIoTfleetwiseAccessTimestreamPermissionsPolicy* con il nome della politica che stai creando.
- Sostituisci *permissions-policy* con il nome del file JSON che contiene la politica di autorizzazione.

```
aws iam create-policy --policy-name AWSIoTfleetwiseAccessTimestreamPermissionsPolicy --
policy-document file://permissions-policy.json
```

```

{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "timestreamIngestion",
 "Effect": "Allow",
 "Action": [
 "timestream:WriteRecords",
 "timestream:Select",
 "timestream:DescribeTable"
],
 "Resource": "table-arn"
 },
 {
 "Sid": "timestreamDescribeEndpoint",
 "Effect": "Allow",
 "Action": [
 "timestream:DescribeEndpoints"
]
 }
]
}

```

```
],
 "Resource": "*"
 }
]
}
```

Per allegare la politica delle autorizzazioni al tuo ruolo IAM

1. Dall'output, copia l'Amazon Resource Name (ARN) della politica delle autorizzazioni.
2. Per collegare la policy di autorizzazione IAM al tuo ruolo IAM, esegui il comando seguente.
  - Sostituisci *permissions-policy-arn* con l'ARN che hai copiato nel passaggio precedente.
  - Sostituiscilo *TimestreamExecutionRole* con il nome del ruolo IAM che hai creato.

```
aws iam attach-role-policy --policy-arn permissions-policy-arn --role-
name TimestreamExecutionRole
```

Per ulteriori informazioni, consulta la sezione [Gestione degli accessi alle AWS risorse nella Guida per l'utente IAM](#).

## Identity and Access Management per AWS IoT FleetWise

AWS Identity and Access Management (IAM) è uno strumento Servizio AWS che aiuta un amministratore a controllare in modo sicuro l'accesso alle AWS risorse. Gli amministratori IAM controllano chi può essere autenticato (effettuato l'accesso) e autorizzato (disporre delle autorizzazioni) a utilizzare le risorse IoT AWS . FleetWise IAM è uno Servizio AWS strumento che puoi utilizzare senza costi aggiuntivi.

### Argomenti

- [Destinatari](#)
- [Autenticazione con identità](#)
- [Gestione dell'accesso con policy](#)
- [Come FleetWise funziona AWS l'IoT con IAM](#)
- [Esempi di policy basate sull'identità per l'IoT AWS FleetWise](#)

- [Risoluzione dei problemi relativi all' FleetWise identità e all'accesso all' AWS IoT](#)

## Destinatari

Il modo in cui utilizzi AWS Identity and Access Management (IAM) varia a seconda del lavoro svolto nell' AWS IoT FleetWise.

**Utente del servizio:** se utilizzi il FleetWise servizio AWS IoT per svolgere il tuo lavoro, l'amministratore ti fornisce le credenziali e le autorizzazioni necessarie. Man mano che utilizzi più FleetWise funzionalità AWS IoT per svolgere il tuo lavoro, potresti aver bisogno di autorizzazioni aggiuntive. La comprensione della gestione dell'accesso ti consente di richiedere le autorizzazioni corrette all'amministratore. Se non riesci ad accedere a una funzionalità AWS dell'IoT FleetWise, consulta [Risoluzione dei problemi relativi all' FleetWise identità e all'accesso all' AWS IoT](#).

**Amministratore del servizio:** se sei responsabile delle FleetWise risorse AWS IoT della tua azienda, probabilmente hai pieno accesso all' AWS IoT FleetWise. È tuo compito determinare a quali FleetWise funzionalità e risorse AWS IoT devono accedere gli utenti del servizio. Devi inviare le richieste all'amministratore IAM per cambiare le autorizzazioni degli utenti del servizio. Esamina le informazioni contenute in questa pagina per comprendere i concetti di base relativi a IAM. Per saperne di più su come la tua azienda può utilizzare IAM con l' AWS IoT FleetWise, consulta [Come FleetWise funziona AWS l'IoT con IAM](#).

**Amministratore IAM:** se sei un amministratore IAM, potresti voler conoscere i dettagli su come scrivere policy per gestire l'accesso all' AWS IoT FleetWise. Per visualizzare esempi di policy FleetWise basate sull'identità AWS IoT che puoi utilizzare in IAM, consulta. [Esempi di policy basate sull'identità per l'IoT AWS FleetWise](#)

## Autenticazione con identità

L'autenticazione è il modo in cui accedi AWS utilizzando le tue credenziali di identità. Devi essere autenticato (aver effettuato l' Utente root dell'account AWS accesso AWS) come utente IAM o assumendo un ruolo IAM.

Puoi accedere AWS come identità federata utilizzando le credenziali fornite tramite una fonte di identità. AWS IAM Identity Center Gli utenti (IAM Identity Center), l'autenticazione Single Sign-On della tua azienda e le tue credenziali di Google o Facebook sono esempi di identità federate. Se accedi come identità federata, l'amministratore ha configurato in precedenza la federazione delle identità utilizzando i ruoli IAM. Quando accedi AWS utilizzando la federazione, assumi indirettamente un ruolo.

A seconda del tipo di utente, puoi accedere al AWS Management Console o al portale di AWS accesso. Per ulteriori informazioni sull'accesso a AWS, vedi [Come accedere al tuo Account AWS nella Guida per l'Accedi ad AWS utente](#).

Se accedi a AWS livello di codice, AWS fornisce un kit di sviluppo software (SDK) e un'interfaccia a riga di comando (CLI) per firmare crittograficamente le tue richieste utilizzando le tue credenziali. Se non utilizzi AWS strumenti, devi firmare tu stesso le richieste. Per ulteriori informazioni sull'utilizzo del metodo consigliato per firmare autonomamente le richieste, consulta [Signing AWS API request](#) nella IAM User Guide.

A prescindere dal metodo di autenticazione utilizzato, potrebbe essere necessario specificare ulteriori informazioni sulla sicurezza. Ad esempio, ti AWS consiglia di utilizzare l'autenticazione a più fattori (MFA) per aumentare la sicurezza del tuo account. Per ulteriori informazioni, consulta [Autenticazione a più fattori](#) nella Guida per l'utente di AWS IAM Identity Center e [Utilizzo dell'autenticazione a più fattori \(MFA\) in AWS](#) nella Guida per l'utente di IAM.

## Account AWS utente root

Quando si crea un account Account AWS, si inizia con un'identità di accesso che ha accesso completo a tutte Servizi AWS le risorse dell'account. Questa identità è denominata utente Account AWS root ed è accessibile effettuando l'accesso con l'indirizzo e-mail e la password utilizzati per creare l'account. Si consiglia vivamente di non utilizzare l'utente root per le attività quotidiane. Conservare le credenziali dell'utente root e utilizzarle per eseguire le operazioni che solo l'utente root può eseguire. Per un elenco completo delle attività che richiedono l'accesso come utente root, consulta la sezione [Attività che richiedono le credenziali dell'utente root](#) nella Guida per l'utente di IAM.

## Identità federata

Come procedura consigliata, richiedi agli utenti umani, compresi gli utenti che richiedono l'accesso come amministratore, di utilizzare la federazione con un provider di identità per accedere Servizi AWS utilizzando credenziali temporanee.

Un'identità federata è un utente dell'elenco utenti aziendale, di un provider di identità Web AWS Directory Service, della directory Identity Center o di qualsiasi utente che accede utilizzando le Servizi AWS credenziali fornite tramite un'origine di identità. Quando le identità federate accedono Account AWS, assumono ruoli e i ruoli forniscono credenziali temporanee.

Per la gestione centralizzata degli accessi, consigliamo di utilizzare AWS IAM Identity Center. Puoi creare utenti e gruppi in IAM Identity Center oppure puoi connetterti e sincronizzarti con un set di

utenti e gruppi nella tua fonte di identità per utilizzarli su tutte le tue applicazioni. Account AWS Per ulteriori informazioni sul Centro identità IAM, consulta [Cos'è Centro identità IAM?](#) nella Guida per l'utente di AWS IAM Identity Center .

## Utenti e gruppi IAM

Un [utente IAM](#) è un'identità interna Account AWS che dispone di autorizzazioni specifiche per una singola persona o applicazione. Ove possibile, consigliamo di fare affidamento a credenziali temporanee invece di creare utenti IAM con credenziali a lungo termine come le password e le chiavi di accesso. Tuttavia, per casi d'uso specifici che richiedono credenziali a lungo termine con utenti IAM, si consiglia di ruotare le chiavi di accesso. Per ulteriori informazioni, consulta la pagina [Rotazione periodica delle chiavi di accesso per casi d'uso che richiedono credenziali a lungo termine](#) nella Guida per l'utente di IAM.

Un [gruppo IAM](#) è un'identità che specifica un insieme di utenti IAM. Non è possibile eseguire l'accesso come gruppo. È possibile utilizzare gruppi per specificare le autorizzazioni per più utenti alla volta. I gruppi semplificano la gestione delle autorizzazioni per set di utenti di grandi dimensioni. Ad esempio, è possibile avere un gruppo denominato IAMAdmins e concedere a tale gruppo le autorizzazioni per amministrare le risorse IAM.

Gli utenti sono diversi dai ruoli. Un utente è associato in modo univoco a una persona o un'applicazione, mentre un ruolo è destinato a essere assunto da chiunque ne abbia bisogno. Gli utenti dispongono di credenziali a lungo termine permanenti, mentre i ruoli forniscono credenziali temporanee. Per ulteriori informazioni, consulta [Quando creare un utente IAM \(invece di un ruolo\)](#) nella Guida per l'utente di IAM.

## Ruoli IAM

Un [ruolo IAM](#) è un'identità interna all'utente Account AWS che dispone di autorizzazioni specifiche. È simile a un utente IAM, ma non è associato a una persona specifica. Puoi assumere temporaneamente un ruolo IAM in AWS Management Console [cambiando ruolo](#). Puoi assumere un ruolo chiamando un'operazione AWS CLI o AWS API o utilizzando un URL personalizzato. Per ulteriori informazioni sui metodi per l'utilizzo dei ruoli, consulta [Utilizzo di ruoli IAM](#) nella Guida per l'utente di IAM.

I ruoli IAM con credenziali temporanee sono utili nelle seguenti situazioni:

- **Accesso utente federato:** per assegnare le autorizzazioni a una identità federata, è possibile creare un ruolo e definire le autorizzazioni per il ruolo. Quando un'identità federata viene

autenticata, l'identità viene associata al ruolo e ottiene le autorizzazioni da esso definite. Per ulteriori informazioni sulla federazione dei ruoli, consulta [Creazione di un ruolo per un provider di identità di terza parte](#) nella Guida per l'utente di IAM. Se utilizzi IAM Identity Center, configura un set di autorizzazioni. IAM Identity Center mette in correlazione il set di autorizzazioni con un ruolo in IAM per controllare a cosa possono accedere le identità dopo l'autenticazione. Per ulteriori informazioni sui set di autorizzazioni, consulta [Set di autorizzazioni](#) nella Guida per l'utente di AWS IAM Identity Center .

- Autorizzazioni utente IAM temporanee: un utente IAM o un ruolo può assumere un ruolo IAM per ottenere temporaneamente autorizzazioni diverse per un'attività specifica.
- Accesso multi-account: è possibile utilizzare un ruolo IAM per permettere a un utente (un principale affidabile) con un account diverso di accedere alle risorse nell'account. I ruoli sono lo strumento principale per concedere l'accesso multi-account. Tuttavia, con alcuni Servizi AWS, è possibile allegare una policy direttamente a una risorsa (anziché utilizzare un ruolo come proxy). Per informazioni sulle differenze tra ruoli e policy basate su risorse per l'accesso multi-account, consulta [Differenza tra i ruoli IAM e le policy basate su risorse](#) nella Guida per l'utente di IAM.
- Accesso a più servizi: alcuni Servizi AWS utilizzano le funzionalità di altri Servizi AWS. Ad esempio, quando effettui una chiamata in un servizio, è comune che tale servizio esegua applicazioni in Amazon EC2 o archivi oggetti in Amazon S3. Un servizio può eseguire questa operazione utilizzando le autorizzazioni dell'entità chiamante, utilizzando un ruolo di servizio o utilizzando un ruolo collegato al servizio.
- Sessioni di accesso diretto (FAS): quando utilizzi un utente o un ruolo IAM per eseguire azioni AWS, sei considerato un preside. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra operazione in un servizio diverso. FAS utilizza le autorizzazioni del principale che chiama un Servizio AWS, combinate con la richiesta Servizio AWS per effettuare richieste ai servizi downstream. Le richieste FAS vengono effettuate solo quando un servizio riceve una richiesta che richiede interazioni con altri Servizi AWS o risorse per essere completata. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le azioni. Per i dettagli delle policy relative alle richieste FAS, consulta la pagina [Forward access sessions](#).
- Ruolo di servizio: un ruolo di servizio è un [ruolo IAM](#) che un servizio assume per eseguire azioni per tuo conto. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione [Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS](#) nella Guida per l'utente di IAM.
- Ruolo collegato al servizio: un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un Servizio AWS. Il servizio può assumere il ruolo per eseguire un'operazione per tuo conto. I

ruoli collegati al servizio vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non modificarle.

- Applicazioni in esecuzione su Amazon EC2: puoi utilizzare un ruolo IAM per gestire le credenziali temporanee per le applicazioni in esecuzione su un'istanza EC2 e che AWS CLI effettuano richieste API. AWS Ciò è preferibile all'archiviazione delle chiavi di accesso nell'istanza EC2. Per assegnare un AWS ruolo a un'istanza EC2 e renderlo disponibile per tutte le sue applicazioni, crei un profilo di istanza collegato all'istanza. Un profilo dell'istanza contiene il ruolo e consente ai programmi in esecuzione sull'istanza EC2 di ottenere le credenziali temporanee. Per ulteriori informazioni, consulta [Utilizzo di un ruolo IAM per concedere autorizzazioni ad applicazioni in esecuzione su istanze di Amazon EC2](#) nella Guida per l'utente di IAM.

Per informazioni sull'utilizzo dei ruoli IAM, consulta [Quando creare un ruolo IAM \(invece di un utente\)](#) nella Guida per l'utente di IAM.

## Gestione dell'accesso con policy

Puoi controllare l'accesso AWS creando policy e collegandole a AWS identità o risorse. Una policy è un oggetto AWS che, se associato a un'identità o a una risorsa, ne definisce le autorizzazioni. AWS valuta queste politiche quando un principale (utente, utente root o sessione di ruolo) effettua una richiesta. Le autorizzazioni nelle policy determinano l'approvazione o il rifiuto della richiesta. La maggior parte delle politiche viene archiviata AWS come documenti JSON. Per ulteriori informazioni sulla struttura e sui contenuti dei documenti delle policy JSON, consulta [Panoramica delle policy JSON](#) nella Guida per l'utente di IAM.

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Per concedere agli utenti l'autorizzazione a eseguire azioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM. Successivamente l'amministratore può aggiungere le policy IAM ai ruoli e gli utenti possono assumere i ruoli.

Le policy IAM definiscono le autorizzazioni relative a un'azione, a prescindere dal metodo utilizzato per eseguirla. Ad esempio, supponiamo di disporre di una policy che consente l'azione `iam:GetRole`. Un utente con tale policy può ottenere informazioni sul ruolo dall' AWS Management Console AWS CLI, dall' AWS CLI, dall' AWS API.

## Policy basate su identità

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruoli IAM). Tali policy definiscono le azioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta [Creazione di policy IAM](#) nella Guida per l'utente di IAM.

Le policy basate su identità possono essere ulteriormente classificate come policy inline o policy gestite. Le policy inline sono integrate direttamente in un singolo utente, gruppo o ruolo. Le politiche gestite sono politiche autonome che puoi allegare a più utenti, gruppi e ruoli nel tuo Account AWS. Le politiche gestite includono politiche AWS gestite e politiche gestite dai clienti. Per informazioni su come scegliere tra una policy gestita o una policy inline, consulta [Scelta fra policy gestite e policy inline](#) nella Guida per l'utente di IAM.

## Policy basate su risorse

Le policy basate su risorse sono documenti di policy JSON che è possibile collegare a una risorsa. Gli esempi più comuni di policy basate su risorse sono le policy di attendibilità dei ruoli IAM e le policy dei bucket Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarle per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le azioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario [specificare un principale](#) in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o Servizi AWS

Le policy basate sulle risorse sono policy inline che si trovano in tale servizio. Non puoi utilizzare le policy AWS gestite di IAM in una policy basata sulle risorse.

## Liste di controllo degli accessi (ACL)

Le liste di controllo degli accessi (ACL) controllano quali principali (membri, utenti o ruoli dell'account) hanno le autorizzazioni per accedere a una risorsa. Le ACL sono simili alle policy basate su risorse, sebbene non utilizzino il formato del documento di policy JSON.

Amazon S3 e Amazon VPC sono esempi di servizi che supportano gli ACL. AWS WAF Per maggiori informazioni sulle ACL, consulta [Panoramica delle liste di controllo degli accessi \(ACL\)](#) nella Guida per gli sviluppatori di Amazon Simple Storage Service.

## Altri tipi di policy

AWS supporta tipi di policy aggiuntivi e meno comuni. Questi tipi di policy possono impostare il numero massimo di autorizzazioni concesse dai tipi di policy più comuni.

- **Limiti delle autorizzazioni:** un limite delle autorizzazioni è una funzione avanzata nella quale si imposta il numero massimo di autorizzazioni che una policy basata su identità può concedere a un'entità IAM (utente o ruolo IAM). È possibile impostare un limite delle autorizzazioni per un'entità. Le autorizzazioni risultanti sono l'intersezione delle policy basate su identità dell'entità e i relativi limiti delle autorizzazioni. Le policy basate su risorse che specificano l'utente o il ruolo nel campo `Principal` sono condizionate dal limite delle autorizzazioni. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni sui limiti delle autorizzazioni, consulta [Limiti delle autorizzazioni per le entità IAM](#) nella Guida per l'utente di IAM.
- **Politiche di controllo dei servizi (SCP):** le SCP sono politiche JSON che specificano le autorizzazioni massime per un'organizzazione o un'unità organizzativa (OU) in AWS Organizations. AWS Organizations è un servizio per il raggruppamento e la gestione centralizzata di più Account AWS di proprietà dell'azienda. Se abiliti tutte le funzionalità in un'organizzazione, puoi applicare le policy di controllo dei servizi (SCP) a uno o tutti i tuoi account. L'SCP limita le autorizzazioni per le entità negli account dei membri, inclusa ciascuna. Utente root dell'account AWS Per ulteriori informazioni su organizzazioni e policy SCP, consulta la pagina sulle [Policy di controllo dei servizi](#) nella Guida per l'utente di AWS Organizations .
- **Policy di sessione:** le policy di sessione sono policy avanzate che vengono trasmesse come parametro quando si crea in modo programmatico una sessione temporanea per un ruolo o un utente federato. Le autorizzazioni della sessione risultante sono l'intersezione delle policy basate su identità del ruolo o dell'utente e le policy di sessione. Le autorizzazioni possono anche provenire da una policy basata su risorse. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni, consulta [Policy di sessione](#) nella Guida per l'utente di IAM.

## Più tipi di policy

Quando più tipi di policy si applicano a una richiesta, le autorizzazioni risultanti sono più complicate da comprendere. Per sapere come si AWS determina se consentire una richiesta quando sono coinvolti più tipi di policy, consulta [Logica di valutazione delle policy](#) nella IAM User Guide.

## Come FleetWise funziona AWS IoT con IAM

Prima di utilizzare IAM per gestire l'accesso all' AWS IoT FleetWise, scopri quali funzionalità IAM sono disponibili per l'uso con l' AWS IoT FleetWise.

Funzionalità IAM che puoi utilizzare con l' AWS IoT FleetWise

| Funzionalità IAM                                       | AWS FleetWise Supporto IoT |
|--------------------------------------------------------|----------------------------|
| <a href="#">Policy basate su identità</a>              | Sì                         |
| <a href="#">Policy basate su risorse</a>               | No                         |
| <a href="#">Azioni di policy</a>                       | Sì                         |
| <a href="#">Risorse relative alle policy</a>           | Sì                         |
| <a href="#">Chiavi di condizione delle policy</a>      | Sì                         |
| <a href="#">Liste di controllo degli accessi (ACL)</a> | No                         |
| <a href="#">ABAC (tag nelle policy)</a>                | Parziale                   |
| <a href="#">Credenziali temporanee</a>                 | Sì                         |
| <a href="#">Autorizzazioni del principale</a>          | Sì                         |
| <a href="#">Ruoli di servizio</a>                      | No                         |
| <a href="#">Ruoli collegati al servizio</a>            | No                         |

Per avere una visione di alto livello di come l' AWS IoT FleetWise e gli altri AWS servizi funzionano con la maggior parte delle funzionalità IAM, consulta [AWS i servizi che funzionano con IAM nella IAM User Guide](#).

### Policy basate sull'identità per l'IoT AWS FleetWise

|                                       |    |
|---------------------------------------|----|
| Supporta le policy basate su identità | Sì |
|---------------------------------------|----|

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le azioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta [Creazione di policy IAM](#) nella Guida per l'utente di IAM.

Con le policy basate su identità di IAM, è possibile specificare quali operazioni e risorse sono consentite o respinte, nonché le condizioni in base alle quali le operazioni sono consentite o respinte. Non è possibile specificare l'entità principale in una policy basata sull'identità perché si applica all'utente o al ruolo a cui è associato. Per informazioni su tutti gli elementi utilizzabili in una policy JSON, consulta [Guida di riferimento agli elementi delle policy JSON IAM](#) nella Guida per l'utente di IAM.

Esempi di policy basate sull'identità per l'IoT AWS FleetWise

Per visualizzare esempi di policy FleetWise basate sull'identità AWS IoT, vedere. [Esempi di policy basate sull'identità per l'IoT AWS FleetWise](#)

Policy basate sulle risorse nell'ambito dell'IoT AWS FleetWise

Supporta le policy basate su risorse

No

Le policy basate su risorse sono documenti di policy JSON che è possibile collegare a una risorsa. Gli esempi più comuni di policy basate su risorse sono le policy di attendibilità dei ruoli IAM e le policy dei bucket Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarle per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le azioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario [specificare un principale](#) in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o. Servizi AWS

Per consentire l'accesso multi-account, puoi specificare un intero account o entità IAM in un altro account come principale in una policy basata sulle risorse. L'aggiunta di un principale multi-account a una policy basata sulle risorse rappresenta solo una parte della relazione di trust. Quando il principale e la risorsa sono diversi Account AWS, un amministratore IAM dell'account affidabile deve inoltre concedere all'entità principale (utente o ruolo) l'autorizzazione ad accedere alla risorsa. L'autorizzazione viene concessa collegando all'entità una policy basata sull'identità. Tuttavia, se una policy basata su risorse concede l'accesso a un principale nello stesso account, non sono richieste

ulteriori policy basate su identità. Per ulteriori informazioni, consulta [Differenza tra i ruoli IAM e le policy basate su risorse](#) nella Guida per l'utente di IAM.

## Azioni politiche per l' AWS IoT FleetWise

Supporta le operazioni di policy

Sì

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire azioni su quali risorse, e in quali condizioni.

L'elemento `Action` di una policy JSON descrive le operazioni che è possibile utilizzare per consentire o negare l'accesso a un criterio. Le azioni politiche in genere hanno lo stesso nome dell'operazione AWS API associata. Ci sono alcune eccezioni, ad esempio le azioni di sola autorizzazione che non hanno un'operazione API corrispondente. Esistono anche alcune operazioni che richiedono più operazioni in una policy. Queste operazioni aggiuntive sono denominate operazioni dipendenti.

Includi le operazioni in una policy per concedere le autorizzazioni a eseguire l'operazione associata.

Per visualizzare un elenco di FleetWise azioni AWS IoT, consulta [Azioni definite dall' AWS IoT FleetWise](#) nel Service Authorization Reference.

Le azioni politiche in AWS IoT FleetWise utilizzano il seguente prefisso prima dell'azione:

```
iotfleetwise
```

Per specificare più operazioni in una sola istruzione, occorre separarle con la virgola.

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

È possibile specificare più azioni tramite caratteri jolly (\*). Ad esempio, per specificare tutte le azioni che iniziano con la parola `List`, includi la seguente azione:

```
"Action": "iotfleetwise:List*"
```

Per visualizzare esempi di policy FleetWise basate sull'identità AWS IoT, vedere. [Esempi di policy basate sull'identità per l'IoT AWS FleetWise](#)

## Risorse politiche per l' AWS IoT FleetWise

|                               |    |
|-------------------------------|----|
| Supporta le risorse di policy | Si |
|-------------------------------|----|

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire operazioni su quali risorse, e in quali condizioni.

L'elemento JSON `Resource` della policy specifica l'oggetto o gli oggetti ai quali si applica l'azione. Le istruzioni devono includere un elemento `Resource` o un elemento `NotResource`. Come best practice, specifica una risorsa utilizzando il suo [nome della risorsa Amazon \(ARN\)](#). Puoi eseguire questa operazione per azioni che supportano un tipo di risorsa specifico, note come autorizzazioni a livello di risorsa.

Per le azioni che non supportano le autorizzazioni a livello di risorsa, ad esempio le operazioni di elenco, utilizza un carattere jolly (\*) per indicare che l'istruzione si applica a tutte le risorse.

```
"Resource": "*"
```

Per visualizzare un elenco dei tipi di FleetWise risorse AWS IoT e dei relativi ARN, consulta [Resources Defined by AWS IoT FleetWise](#) nel Service Authorization Reference. Per sapere con quali azioni è possibile specificare l'ARN di ciascuna risorsa, consulta [Azioni definite dall'IoT AWS FleetWise](#)

Per visualizzare esempi di policy FleetWise basate sull'identità AWS IoT, vedere. [Esempi di policy basate sull'identità per l'IoT AWS FleetWise](#)

## Chiavi delle condizioni delle policy per l' AWS IoT FleetWise

|                                                                       |    |
|-----------------------------------------------------------------------|----|
| Supporta le chiavi di condizione delle policy specifiche del servizio | Si |
|-----------------------------------------------------------------------|----|

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire azioni su quali risorse, e in quali condizioni.

L'elemento `Condition` (o blocco `Condition`) consente di specificare le condizioni in cui un'istruzione è in vigore. L'elemento `Condition` è facoltativo. Puoi compilare espressioni condizionali che utilizzano [operatori di condizione](#), ad esempio uguale a o minore di, per soddisfare la condizione nella policy con i valori nella richiesta.

Se specifichi più elementi `Condition` in un'istruzione o più chiavi in un singolo elemento `Condition`, questi vengono valutati da AWS utilizzando un'operazione AND logica. Se si specificano più valori per una singola chiave di condizione, AWS valuta la condizione utilizzando un'operazione logica. OR Tutte le condizioni devono essere soddisfatte prima che le autorizzazioni dell'istruzione vengano concesse.

Puoi anche utilizzare variabili segnaposto quando specifichi le condizioni. Ad esempio, puoi autorizzare un utente IAM ad accedere a una risorsa solo se è stata taggata con il relativo nome utente IAM. Per ulteriori informazioni, consulta [Elementi delle policy IAM: variabili e tag](#) nella Guida per l'utente di IAM.

AWS supporta chiavi di condizione globali e chiavi di condizione specifiche del servizio. Per visualizzare tutte le chiavi di condizione AWS globali, consulta le chiavi di [contesto delle condizioni AWS globali nella Guida](#) per l'utente IAM.

Per visualizzare un elenco di chiavi di FleetWise condizione AWS IoT, consulta [Condition Keys for AWS IoT FleetWise](#) nel Service Authorization Reference. Per sapere con quali azioni e risorse puoi utilizzare una chiave di condizione, vedi [Azioni definite dall' AWS IoT FleetWise](#) .

Per visualizzare esempi di policy FleetWise basate sull'identità AWS IoT, vedere. [Esempi di policy basate sull'identità per l'IoT AWS FleetWise](#)

## Liste di controllo degli accessi (ACL) nell' AWS IoT FleetWise

Supporta le ACL

No

Le liste di controllo degli accessi (ACL) controllano quali principali (membri, utenti o ruoli dell'account) hanno le autorizzazioni ad accedere a una risorsa. Le ACL sono simili alle policy basate su risorse, sebbene non utilizzino il formato del documento di policy JSON.

## Controllo degli accessi basato sugli attributi (ABAC) con IoT AWS FleetWise

Supporta ABAC (tag nelle policy)

Parziale

Il controllo dell'accesso basato su attributi (ABAC) è una strategia di autorizzazione che definisce le autorizzazioni in base agli attributi. In AWS, questi attributi sono chiamati tag. Puoi allegare tag a entità IAM (utenti o ruoli) e a molte AWS risorse. L'assegnazione di tag alle entità e alle risorse è il primo passaggio di ABAC. In seguito, vengono progettate policy ABAC per consentire operazioni quando il tag dell'entità principale corrisponde al tag sulla risorsa a cui si sta provando ad accedere.

La strategia ABAC è utile in ambienti soggetti a una rapida crescita e aiuta in situazioni in cui la gestione delle policy diventa impegnativa.

Per controllare l'accesso basato su tag, fornisci informazioni sui tag nell'[elemento condizione](#) di una policy utilizzando le chiavi di condizione `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`.

Se un servizio supporta tutte e tre le chiavi di condizione per ogni tipo di risorsa, il valore per il servizio è Yes (Sì). Se un servizio supporta tutte e tre le chiavi di condizione solo per alcuni tipi di risorsa, allora il valore sarà Parziale.

Per ulteriori informazioni su ABAC, consulta [Che cos'è ABAC?](#) nella Guida per l'utente di IAM. Per visualizzare un tutorial con i passaggi per l'impostazione di ABAC, consulta [Utilizzo del controllo degli accessi basato su attributi \(ABAC\)](#) nella Guida per l'utente di IAM.

#### Note

AWS Supporta FleetWise solo l'`IoTiam:PassRole`, necessario per il funzionamento dell'`CreateCampaignAPI`.

## Utilizzo di credenziali temporanee con IoT AWS FleetWise

|                                    |    |
|------------------------------------|----|
| Supporta le credenziali temporanee | Sì |
|------------------------------------|----|

Alcune Servizi AWS non funzionano quando si accede utilizzando credenziali temporanee. Per ulteriori informazioni, incluse quelle che Servizi AWS funzionano con credenziali temporanee, consulta la sezione relativa alla [Servizi AWS compatibilità con IAM nella IAM User Guide](#).

Stai utilizzando credenziali temporanee se accedi AWS Management Console utilizzando qualsiasi metodo tranne nome utente e password. Ad esempio, quando accedi AWS utilizzando il link Single

Sign-On (SSO) della tua azienda, tale processo crea automaticamente credenziali temporanee. Le credenziali temporanee vengono create in automatico anche quando accedi alla console come utente e poi cambi ruolo. Per ulteriori informazioni sullo scambio dei ruoli, consulta [Cambio di un ruolo \(console\)](#) nella Guida per l'utente di IAM.

È possibile creare manualmente credenziali temporanee utilizzando l'API o AWS CLI. AWS consiglia di generare dinamicamente credenziali temporanee anziché utilizzare chiavi di accesso a lungo termine. Per ulteriori informazioni, consulta [Credenziali di sicurezza provvisorie in IAM](#).

## Autorizzazioni principali multiservizio per l'IoT AWS FleetWise

|                                                    |    |
|----------------------------------------------------|----|
| Supporta l'inoltro delle sessioni di accesso (FAS) | Si |
|----------------------------------------------------|----|

Quando utilizzi un utente o un ruolo IAM per eseguire azioni AWS, sei considerato un principale. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra operazione in un servizio diverso. FAS utilizza le autorizzazioni del principale che chiama un Servizio AWS, in combinazione con la richiesta Servizio AWS per effettuare richieste ai servizi downstream. Le richieste FAS vengono effettuate solo quando un servizio riceve una richiesta che richiede interazioni con altri Servizi AWS o risorse per essere completata. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le azioni. Per i dettagli delle policy relative alle richieste FAS, consulta la pagina [Forward access sessions](#).

## Ruoli di servizio per l' AWS IoT FleetWise

|                              |    |
|------------------------------|----|
| Supporta i ruoli di servizio | No |
|------------------------------|----|

Un ruolo di servizio è un [ruolo IAM](#) che un servizio assume per eseguire operazioni per tuo conto. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione [Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS](#) nella Guida per l'utente di IAM.

**⚠ Warning**

La modifica delle autorizzazioni per un ruolo di servizio potrebbe interrompere la FleetWise funzionalità AWS IoT. Modifica i ruoli di servizio solo quando l' AWS IoT FleetWise fornisce indicazioni in tal senso.

## Ruoli collegati ai servizi per l'IoT AWS FleetWise

Supporta i ruoli collegati ai servizi

No

Un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un servizio AWS. Il servizio può assumere il ruolo per eseguire un'operazione per tuo conto. I ruoli collegati al servizio vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non modificarle.

Per ulteriori informazioni su come creare e gestire i ruoli collegati ai servizi, consulta [Servizi AWS supportati da IAM](#). Trova un servizio nella tabella che include un Yes nella colonna Service-linked role (Ruolo collegato ai servizi). Scegli il collegamento Sì per visualizzare la documentazione relativa al ruolo collegato ai servizi per tale servizio.

## Utilizzo di ruoli collegati ai servizi per AWS IoT FleetWise

AWS IoT FleetWise utilizza AWS Identity and Access Management ruoli [collegati ai servizi](#) (IAM). Un ruolo collegato ai servizi è un tipo unico di ruolo IAM collegato direttamente ad AWS IoT. FleetWise I ruoli collegati ai servizi sono predefiniti da AWS IoT FleetWise e includono le autorizzazioni necessarie ad AWS IoT per inviare FleetWise metriche ad Amazon. CloudWatch Per ulteriori informazioni, consulta [Monitoraggio AWS dell'IoT FleetWise con Amazon CloudWatch](#).

Un ruolo collegato ai servizi semplifica la configurazione di AWS FleetWise IoT perché non è necessario aggiungere manualmente le autorizzazioni necessarie. AWS IoT FleetWise definisce le autorizzazioni dei suoi ruoli collegati ai servizi e, se non diversamente definito, solo AWS IoT FleetWise può assumerne i ruoli. Le autorizzazioni definite includono policy di attendibilità e di autorizzazioni. Questa policy di autorizzazione non può essere associata a nessun'altra entità IAM.

È possibile eliminare un ruolo collegato ai servizi solo dopo aver eliminato le risorse correlate. Questo protegge le tue FleetWise risorse AWS IoT perché non puoi rimuovere inavvertitamente l'autorizzazione ad accedere alle risorse.

Per informazioni su altri servizi che supportano ruoli collegati ai servizi, consulta [AWS i servizi che funzionano con IAM](#) e cerca i servizi con Sì nella colonna Ruoli collegati ai servizi. Per visualizzare la documentazione relativa al ruolo collegato ai servizi per quel servizio, scegli un Sì con un collegamento.

## Autorizzazioni di ruolo collegate ai servizi per AWS IoT FleetWise

AWS IoT FleetWise utilizza il ruolo collegato ai servizi denominato `AWSServiceRoleForIoT FleetWise`: una policy gestita da AWS che viene utilizzata per tutte le out-of-the-box autorizzazioni per AWS IoT FleetWise.

Il ruolo `AWSServiceRoleForIoT FleetWise` collegato al servizio prevede che i seguenti servizi assumano il ruolo:

- `IoT FleetWise`

La policy di autorizzazione dei ruoli denominata `AWSIoT FleetWiseServiceRolePolicy` consente FleetWise ad AWS IoT di completare le seguenti azioni sulle risorse specificate:

- Azione: `cloudwatch:PutMetricData` sulla risorsa: \*

Per consentire a un'entità IAM (come un utente, un gruppo o un ruolo) di creare, modificare o eliminare un ruolo collegato ai servizi devi configurare le relative autorizzazioni. Per ulteriori informazioni, consulta [Autorizzazioni del ruolo collegato ai servizi](#) nella Guida per l'utente di IAM.

## Creazione di un ruolo collegato ai servizi per AWS IoT FleetWise

Non hai bisogno di creare manualmente un ruolo collegato ai servizi. Quando registri un account nella FleetWise console AWS IoT, o nell' AWS API AWS CLI, AWS IoT FleetWise crea il ruolo collegato al servizio per te. Per ulteriori informazioni, consulta [Configurazione delle impostazioni](#).

## Creazione di un ruolo collegato ai servizi in AWS IoT FleetWise (console)

Non hai bisogno di creare manualmente un ruolo collegato ai servizi. Quando registri un account nella FleetWise console AWS IoT, nella AWS CLI o nell' AWS API, AWS IoT FleetWise crea il ruolo collegato al servizio per te.

## Modifica di un ruolo collegato ai servizi per AWS IoT FleetWise

Non è possibile modificare il ruolo `AWSServiceRoleForIoT FleetWise` collegato al servizio in AWS IoT FleetWise. Poiché varie entità potrebbero fare riferimento a qualsiasi ruolo collegato al servizio che

crei, non puoi modificare il nome del ruolo. È possibile tuttavia modificarne la descrizione utilizzando IAM. Per ulteriori informazioni, consulta [Modifica di un ruolo collegato ai servizi](#) nella Guida per l'utente di IAM.

## Pulizia di un ruolo collegato ai servizi

Prima di utilizzare IAM; per eliminare un ruolo collegato al servizio, è necessario prima rimuovere qualsiasi risorsa utilizzata dal ruolo.

### Note

Se FleetWise AWS IoT utilizza il ruolo quando si tenta di eliminare le risorse, l'eliminazione potrebbe non riuscire. In questo caso, attendi alcuni minuti e quindi ripeti l'operazione. Per scoprire come eliminarli service-linked-role tramite la console, la AWS CLI o l' AWS API, consulta [Using service-linked roles](#) nella IAM User Guide.

Se elimini questo ruolo collegato al servizio e poi devi crearlo di nuovo, puoi registrare un account con AWS IoT. FleetWise AWS IoT crea FleetWise quindi nuovamente il ruolo collegato ai servizi per te.

## Esempi di policy basate sull'identità per l'IoT AWS FleetWise

Per impostazione predefinita, gli utenti e i ruoli non dispongono dell'autorizzazione per creare o modificare FleetWise risorse AWS IoT. Inoltre, non possono eseguire attività utilizzando AWS Management Console, AWS Command Line Interface (AWS CLI) o AWS l'API. Per concedere agli utenti l'autorizzazione a eseguire azioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM. L'amministratore può quindi aggiungere le policy IAM ai ruoli e gli utenti possono assumere i ruoli.

Per informazioni su come creare una policy basata su identità IAM utilizzando questi documenti di policy JSON di esempio, consulta [Creazione di policy IAM](#) nella Guida per l'utente di IAM.

Per informazioni dettagliate sulle azioni e sui tipi di risorse definiti dall' AWS IoT FleetWise, incluso il formato degli ARN per ciascun tipo di risorsa, consulta [Actions, Resources and Condition Keys for AWS IoT FleetWise](#) nel Service Authorization Reference.

### Argomenti

- [Best practice per le policy](#)

- [Utilizzo della FleetWise console AWS IoT](#)
- [Consentire agli utenti di visualizzare le loro autorizzazioni](#)
- [Accedi alle risorse in Amazon Timestream](#)

## Best practice per le policy

Le policy basate sull'identità determinano se qualcuno può creare, accedere o eliminare FleetWise risorse AWS IoT nel tuo account. Queste azioni possono comportare costi aggiuntivi per l'Account AWS. Quando crei o modifichi policy basate su identità, segui queste linee guida e raccomandazioni:

- Inizia con le policy AWS gestite e passa alle autorizzazioni con privilegi minimi: per iniziare a concedere autorizzazioni a utenti e carichi di lavoro, utilizza le politiche gestite che concedono le autorizzazioni per molti casi d'uso comuni. AWS Sono disponibili nel tuo Account AWS. Ti consigliamo di ridurre ulteriormente le autorizzazioni definendo politiche gestite dai AWS clienti specifiche per i tuoi casi d'uso. Per ulteriori informazioni, consulta [Policy gestite da AWS](#) o [Policy gestite da AWS per le funzioni dei processi](#) nella Guida per l'utente IAM.
- Applica le autorizzazioni con privilegi minimi: quando imposti le autorizzazioni con le policy IAM, concedi solo le autorizzazioni richieste per eseguire un'attività. Puoi farlo definendo le azioni che possono essere intraprese su risorse specifiche in condizioni specifiche, note anche come autorizzazioni con privilegi minimi. Per ulteriori informazioni sull'utilizzo di IAM per applicare le autorizzazioni, consulta [Policy e autorizzazioni in IAM](#) nella Guida per l'utente di IAM.
- Condizioni d'uso nelle policy IAM per limitare ulteriormente l'accesso: per limitare l'accesso ad azioni e risorse puoi aggiungere una condizione alle tue policy. Ad esempio, è possibile scrivere una condizione di policy per specificare che tutte le richieste devono essere inviate utilizzando SSL. Puoi anche utilizzare le condizioni per concedere l'accesso alle azioni del servizio se vengono utilizzate tramite uno specifico Servizio AWS, ad esempio AWS CloudFormation. Per ulteriori informazioni, consulta la sezione [Elementi delle policy JSON di IAM: condizione](#) nella Guida per l'utente di IAM.
- Utilizzo di IAM Access Analyzer per convalidare le policy IAM e garantire autorizzazioni sicure e funzionali: IAM Access Analyzer convalida le policy nuove ed esistenti in modo che aderiscano alla sintassi della policy IAM (JSON) e alle best practice di IAM. IAM Access Analyzer offre oltre 100 controlli delle policy e consigli utili per creare policy sicure e funzionali. Per ulteriori informazioni, consulta [Convalida delle policy per IAM Access Analyzer](#) nella Guida per l'utente di IAM.
- Richiedi l'autenticazione a più fattori (MFA): se hai uno scenario che richiede utenti IAM o un utente root nel Account AWS tuo, attiva l'MFA per una maggiore sicurezza. Per richiedere la MFA

quando vengono chiamate le operazioni API, aggiungi le condizioni MFA alle policy. Per ulteriori informazioni, consulta [Configurazione dell'accesso alle API protetto con MFA](#) nella Guida per l'utente di IAM.

Per maggiori informazioni sulle best practice in IAM, consulta [Best practice di sicurezza in IAM](#) nella Guida per l'utente di IAM.

## Utilizzo della FleetWise console AWS IoT

Per accedere alla FleetWise console AWS IoT, è necessario disporre di un set minimo di autorizzazioni. Queste autorizzazioni devono consentirti di elencare e visualizzare i dettagli sulle FleetWise risorse AWS IoT presenti nel tuo Account AWS. Se crei una policy basata sull'identità più restrittiva rispetto alle autorizzazioni minime richieste, la console non funzionerà nel modo previsto per le entità (utenti o ruoli) associate a tale policy.

Non è necessario consentire autorizzazioni minime per la console per gli utenti che effettuano chiamate solo verso AWS CLI o l' AWS API. Al contrario, concedi l'accesso solo alle operazioni che corrispondono all'operazione API che stanno cercando di eseguire.

Per garantire che utenti e ruoli possano ancora utilizzare la FleetWise console AWS IoT, collega anche l' AWS IoT FleetWise ConsoleAccess o la policy ReadOnly AWS gestita alle entità. Per ulteriori informazioni, consulta [Aggiunta di autorizzazioni a un utente](#) nella Guida per l'utente IAM.

## Consentire agli utenti di visualizzare le loro autorizzazioni

Questo esempio mostra in che modo è possibile creare una policy che consente agli utenti IAM di visualizzare le policy inline e gestite che sono collegate alla relativa identità utente. Questa policy include le autorizzazioni per completare questa azione sulla console o utilizzando programmaticamente l' AWS CLI API o AWS

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Sid": "ViewOwnUserInfo",
 "Effect": "Allow",
 "Action": [
 "iam:GetUserPolicy",
 "iam:ListGroupsWithUser",

```

```

 "iam:ListAttachedUserPolicies",
 "iam:ListUserPolicies",
 "iam:GetUser"
],
 "Resource": ["arn:aws:iam::*:user/${aws:username}"]
},
{
 "Sid": "NavigateInConsole",
 "Effect": "Allow",
 "Action": [
 "iam:GetGroupPolicy",
 "iam:GetPolicyVersion",
 "iam:GetPolicy",
 "iam:ListAttachedGroupPolicies",
 "iam:ListGroupPolicies",
 "iam:ListPolicyVersions",
 "iam:ListPolicies",
 "iam:ListUsers"
],
 "Resource": "*"
}
]
}

```

## Accedi alle risorse in Amazon Timestream

Prima di utilizzare l' AWS IoT FleetWise, devi registrare il tuo AWS account, IAM e le risorse Amazon Timestream per concedere l'autorizzazione AWS FleetWise IoT a inviare i dati Cloud AWS del veicolo a tuo nome. Per registrarti, devi:

- Un database Amazon Timestream.
- Una tabella creata nel database Amazon Timestream specificato.
- Un ruolo IAM che consente FleetWise all' AWS IoT di inviare dati ad Amazon Timestream.

Per ulteriori informazioni, incluse procedure e politiche di esempio, consulta [Configura](#) le impostazioni.

# Risoluzione dei problemi relativi all' FleetWise identità e all'accesso all' AWS IoT

Utilizza le seguenti informazioni per aiutarti a diagnosticare e risolvere i problemi più comuni che potresti riscontrare quando lavori con AWS IoT FleetWise e IAM.

## Argomenti

- [Non sono autorizzato a eseguire un'azione in AWS IoT FleetWise](#)
- [Non sono autorizzato a eseguire iam: PassRole](#)
- [Voglio consentire a persone esterne a me di accedere Account AWS alle mie FleetWise risorse AWS IoT](#)

## Non sono autorizzato a eseguire un'azione in AWS IoT FleetWise

Se ti AWS Management Console dice che non sei autorizzato a eseguire un'azione, devi contattare l'amministratore per ricevere assistenza. L'amministratore è colui che ti ha fornito le credenziali di accesso.

L'errore di esempio seguente si verifica quando l'utente mateojackson IAM tenta di utilizzare la console per visualizzare i dettagli su una *myVehicle* risorsa fittizia ma non dispone delle `iotfleetwise:GetVehicleStatus` autorizzazioni.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
iotfleetwise:GetVehicleStatus on resource: myVehicle
```

In questo caso, Mateo richiede al suo amministratore di aggiornare le policy per poter accedere alla risorsa *myVehicle* utilizzando l'azione `iotfleetwise:GetVehicleStatus`.

## Non sono autorizzato a eseguire iam: PassRole

Se ricevi un messaggio di errore indicante che non sei autorizzato a eseguire l'`iam:PassRole` azione, le tue policy devono essere aggiornate per consentirti di passare un ruolo all' AWS IoT FleetWise.

Alcuni Servizi AWS consentono di trasferire un ruolo esistente a quel servizio invece di creare un nuovo ruolo di servizio o un ruolo collegato al servizio. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per trasmettere il ruolo al servizio.

Il seguente errore di esempio si verifica quando un utente IAM denominato `marymajor` tenta di utilizzare la console per eseguire un'azione in AWS IoT FleetWise. Tuttavia, l'azione richiede che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per passare il ruolo al servizio.

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

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

## Voglio consentire a persone esterne a me di accedere Account AWS alle mie FleetWise risorse AWS IoT

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle tue risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo. Per servizi che supportano policy basate su risorse o liste di controllo accessi (ACL), utilizza tali policy per concedere alle persone l'accesso alle tue risorse.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- Per sapere se l' AWS IoT FleetWise supporta queste funzionalità, consulta [Come FleetWise funziona AWS l'IoT con IAM](#).
- Per scoprire come fornire l'accesso alle tue risorse attraverso Account AWS le risorse di tua proprietà, consulta [Fornire l'accesso a un utente IAM in un altro Account AWS di tua proprietà](#) nella IAM User Guide.
- Per scoprire come fornire l'accesso alle tue risorse a terze parti Account AWS, consulta [Fornire l'accesso a soggetti Account AWS di proprietà di terze parti](#) nella Guida per l'utente IAM.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consulta [Fornire l'accesso a utenti autenticati esternamente \(Federazione delle identità\)](#) nella Guida per l'utente di IAM.
- Per informazioni sulle differenze tra l'utilizzo di ruoli e policy basate su risorse per l'accesso multi-account, consulta [Differenza tra i ruoli IAM e le policy basate su risorse](#) nella Guida per l'utente IAM.

# Convalida della conformità per l'IoT AWS FleetWise

## Note

AWS L'IoT FleetWise non rientra nell'ambito di alcun programma di AWS conformità.

Per sapere se un Servizio AWS programma rientra nell'ambito di specifici programmi di conformità, consulta Servizi AWS la sezione [Ambito per programma di conformità Servizi AWS](#) di conformità 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 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)).

- [Valutazione delle risorse con regole](#) nella Guida per gli AWS Config sviluppatori: 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 nell'IoT AWS FleetWise

L'infrastruttura AWS globale è costruita attorno a AWS regioni e zone di disponibilità. Le regioni forniscono più zone di disponibilità fisicamente separate e isolate, connesse tramite reti altamente ridondanti, a bassa latenza e throughput elevato. 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 a data center singolo o multiplo tradizionali.

Per ulteriori informazioni su AWS regioni e zone di disponibilità, consulta [Infrastruttura AWS globale](#).

### Note

I dati elaborati dall' AWS IoT FleetWise vengono archiviati in un database Amazon Timestream. Timestream supporta i backup in altre AWS zone o regioni di disponibilità. Tuttavia, puoi scrivere la tua applicazione utilizzando Timestream SDK per interrogare i dati e salvarli nella destinazione di tua scelta. Per ulteriori informazioni su Amazon Timestream, consulta [l'Amazon Timestream Developer Guide](#).

## Sicurezza dell'infrastruttura nell' AWS IoT FleetWise

In quanto servizio gestito, l' AWS IoT FleetWise è protetto dalla sicurezza di rete AWS globale. Per informazioni sui servizi AWS di sicurezza e su come AWS protegge l'infrastruttura, consulta [AWS Cloud Security](#). Per progettare il tuo AWS ambiente utilizzando le migliori pratiche per la sicurezza dell'infrastruttura, vedi [Infrastructure Protection](#) in Security Pillar AWS Well-Architected Framework.

Utilizzi chiamate API AWS pubblicate per accedere all' AWS IoT FleetWise attraverso la rete. I client devono supportare quanto segue:

- Transport Layer Security (TLS). È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Suite di cifratura con Perfect Forward Secrecy (PFS), ad esempio Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Inoltre, le richieste devono essere firmate utilizzando un ID chiave di accesso e una chiave di accesso segreta associata a un principale IAM. O puoi utilizzare [AWS Security Token Service](#) (AWS STS) per generare credenziali di sicurezza temporanee per sottoscrivere le richieste.

È possibile richiamare queste operazioni API da qualsiasi posizione di rete, ma l' AWS IoT FleetWise supporta politiche di accesso basate sulle risorse, che possono includere restrizioni basate sull'indirizzo IP di origine. Puoi anche utilizzare FleetWise le policy AWS IoT per controllare l'accesso da endpoint Amazon Virtual Private Cloud (Amazon VPC) specifici o VPC specifici. In effetti, questo isola l'accesso alla rete a una determinata FleetWise risorsa AWS IoT solo dal VPC specifico all'interno AWS della rete.

### Argomenti

- [Connessione all' AWS IoT FleetWise tramite un'interfaccia VPC endpoint](#)

## Connessione all' AWS IoT FleetWise tramite un'interfaccia VPC endpoint

Puoi connetterti direttamente all' AWS IoT FleetWise utilizzando un'[interfaccia VPC endpoint \(AWS PrivateLink\)](#) nel tuo Virtual Private Cloud (VPC), invece di connetterti tramite Internet. Quando si utilizza un endpoint VPC di interfaccia, la comunicazione tra VPC e AWS IoT FleetWise viene condotta interamente all'interno della rete. AWS Ogni endpoint VPC è rappresentato da una o più [interfacce di rete elastiche \(ENI\)](#) con indirizzi IP privati nelle sottoreti del VPC.

L'interfaccia VPC endpoint collega il tuo VPC direttamente all' AWS FleetWise IoT senza un gateway Internet, un dispositivo NAT, una connessione VPN o una connessione. AWS Direct Connect Le istanze del tuo VPC non necessitano di indirizzi IP pubblici per comunicare con l'API AWS IoT FleetWise.

Per utilizzare l' AWS IoT FleetWise tramite il tuo VPC, devi connetterti da un'istanza che si trova all'interno del VPC o connettere la tua rete privata al tuo VPC utilizzando una (VPN) o. AWS Virtual Private Network AWS Direct Connect Per informazioni su Amazon VPN, consulta [Connessioni VPN](#) nella Guida per l'utente di Amazon Virtual Private Cloud. Per informazioni su AWS Direct Connect, consulta [Creazione di una connessione nella Guida](#) per l'AWS Direct Connect utente.

È possibile creare un endpoint VPC di interfaccia per connettersi all' AWS IoT FleetWise utilizzando i comandi AWS console o AWS Command Line Interface (AWS CLI). Per ulteriori informazioni, consulta [Creazione di un endpoint di interfaccia](#).

Dopo aver creato un endpoint VPC di interfaccia, se abiliti i nomi host DNS privati per l'endpoint, l'endpoint AWS IoT predefinito si risolve nel tuo FleetWise endpoint VPC. L'endpoint del nome di servizio predefinito per AWS IoT FleetWise è nel seguente formato.

```
iotfleetwise.Region.amazonaws.com
```

Se non abiliti i nomi host DNS privati, Amazon VPC fornisce un nome di endpoint DNS che puoi utilizzare nel seguente formato.

```
VPCE_ID.iotfleetwise.Region.vpce.amazonaws.com
```

Per ulteriori informazioni, consulta [Interface VPC endpoints \(AWS PrivateLink\)](#) nella Amazon VPC User Guide.

AWS L'IoT FleetWise supporta l'esecuzione di chiamate a tutte le sue [azioni API](#) all'interno del tuo VPC.

Puoi collegare le policy di endpoint VPC a un endpoint VPC per controllare l'accesso per le entità principali IAM. Puoi inoltre associare i gruppi di sicurezza a un endpoint VPC per controllare l'accesso in ingresso e in uscita in base all'origine e alla destinazione del traffico di rete, ad esempio un intervallo di indirizzi IP. Per ulteriori informazioni, consulta [Controllo dell'accesso ai servizi con endpoint VPC](#).

## Creazione di una policy sugli endpoint VPC per l'IoT AWS FleetWise

Puoi creare una policy per gli endpoint Amazon VPC for AWS IoT FleetWise per specificare quanto segue:

- Il principal che può o non può eseguire operazioni
- Le azioni che possono o non possono essere eseguite

Per ulteriori informazioni, consulta [Controllo degli accessi ai servizi con endpoint VPC](#) nella Guida per l'utente di Amazon VPC.

Example — Policy degli endpoint VPC per negare tutti gli accessi da un account specifico AWS

La seguente policy sugli endpoint VPC nega all' AWS account **123456789012** tutte le chiamate API che utilizzano l'endpoint.

```
{
 "Statement": [
 {
 "Action": "*",
 "Effect": "Allow",
 "Resource": "*",
 "Principal": "*"
 },
 {
 "Action": "*",
 "Effect": "Deny",
 "Resource": "*",
 "Principal": {
 "AWS": [
 "123456789012"
]
 }
 }
]
}
```

Example - Policy di endpoint VPC per consentire l'accesso VPC solo a un'entità principale IAM (utente) specificata

*La seguente politica degli endpoint VPC consente l'accesso completo solo all'utente lijuan nell'account 123456789012. AWS* Nega a tutti gli altri principali IAM l'accesso all'endpoint.

```
{
 "Statement": [
 {
 "Action": "*",
 "Effect": "Allow",
 "Resource": "*",
 "Principal": {
 "AWS": [
 "arn:aws:iam::123456789012:user/lijuan"
]
 }
 }
]
}
```

Example — Policy sugli endpoint VPC per le azioni IoT AWS FleetWise

Di seguito è riportato un esempio di policy sugli endpoint per l' AWS IoT FleetWise. *Se collegata a un endpoint, questa policy concede l'accesso alle FleetWise azioni AWS IoT elencate per l'utente IAM FleetWise nel 123456789012. Account AWS*

```
{
 "Statement": [
 {
 "Principal": {
 "AWS": [
 "arn:aws:iam::123456789012:user/fleetWise"
]
 },
 "Resource": "*",
 "Effect": "Allow",
 "Action": [
 "iotfleetwise:ListFleets",
 "iotfleetwise:ListCampaigns",
 "iotfleetwise:CreateVehicle",
]
 }
]
}
```

```
]
}
```

## Analisi della configurazione e delle vulnerabilità nell'IoT AWS FleetWise

Gli ambienti IoT possono essere costituiti da un numero elevato di dispositivi con funzionalità diverse, usati per lunghi periodi di tempo e distribuiti in varie aree geografiche. Queste caratteristiche rendono la configurazione di un dispositivo complessa e soggetta a errori. Inoltre, poiché i dispositivi sono spesso limitati in termini di potenza di calcolo, memoria e capacità di archiviazione, l'uso della crittografia e di altre forme di sicurezza sui dispositivi è limitato. I dispositivi usano spesso software con vulnerabilità note. Questi fattori rendono i dispositivi IoT, compresi i veicoli che raccolgono dati per l'AWS IoT FleetWise, un bersaglio interessante per gli hacker e rendono difficile proteggerli su base continuativa.

La configurazione e i controlli IT sono una responsabilità condivisa tra voi AWS e voi, i nostri clienti. Per ulteriori informazioni, consulta il [modello di responsabilità AWS condivisa](#).

## Best practice di sicurezza per l'AWS IoT FleetWise

AWS FleetWise L'IoT offre una serie di funzionalità di sicurezza da considerare 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 ulteriori informazioni sulla sicurezza, AWS IoT consulta [le migliori pratiche di sicurezza AWS IoT Core nella Guida per gli AWS IoT sviluppatori](#)

## Concedere autorizzazioni minime possibili

Segui il principio del privilegio minimo utilizzando il set minimo di autorizzazioni nei ruoli IAM. Limita l'uso del \* wildcard per le Resource proprietà Action and nelle tue policy IAM. Invece, dichiarare un insieme finito di operazioni e risorse quando possibile. Per ulteriori informazioni su privilegi minimi e altre best practice sulle policy, consulta [the section called "Best practice per le policy"](#).

## Non registrare informazioni riservate

È necessario impedire la registrazione delle credenziali e di altre informazioni di identificazione personale (PII). Ti consigliamo di implementare le seguenti misure di sicurezza:

- Non utilizzare informazioni sensibili nei nomi dei dispositivi.
- Non utilizzare informazioni sensibili nei nomi e negli ID delle FleetWise risorse AWS IoT, ad esempio nei nomi delle campagne, nei manifesti dei decoder, nei modelli di veicoli e nei cataloghi dei segnali o negli ID di veicoli e flotte.

## Utilizzalo per visualizzare la cronologia AWS CloudTrail delle chiamate API

Puoi visualizzare una cronologia delle chiamate FleetWise API AWS IoT effettuate sul tuo account per scopi di analisi della sicurezza e risoluzione dei problemi operativi. Per ricevere una cronologia delle chiamate all' FleetWise API AWS IoT effettuate sul tuo account, è sufficiente attivare AWS Management Console. CloudTrail Per ulteriori informazioni, consulta [the section called “Log CloudTrail”](#).

## Tenere sincronizzato l'orologio del dispositivo

È importante avere un orario preciso sul dispositivo. I certificati X.509 hanno data e ora di scadenza. L'orologio sul dispositivo viene utilizzato per verificare che un certificato server sia ancora valido. Gli orologi dei dispositivi possono andare alla deriva nel tempo o le batterie possono scaricarsi.

Per ulteriori informazioni, consulta la best practice [Tenere sincronizzato l'orologio del dispositivo](#) nella Guida per sviluppatori AWS IoT Core .

# Monitoraggio dell'AWS IoT FleetWise

Il monitoraggio è una parte importante per mantenere l'affidabilità, la disponibilità e le prestazioni dell'AWS IoT FleetWise e delle altre AWS soluzioni. AWS fornisce i seguenti strumenti di monitoraggio per monitorare l'AWS IoT FleetWise, segnalare quando qualcosa non va e intraprendere azioni automatiche quando appropriato:

- Amazon CloudWatch monitora AWS le tue risorse e le applicazioni su cui esegui AWS in tempo reale. Puoi raccogliere i parametri e tenerne traccia, creare pannelli di controllo personalizzati e impostare allarmi che inviino una notifica o intraprendano azioni quando un parametro specificato raggiunge una determinata soglia. Ad esempio, puoi tenere CloudWatch traccia dell'utilizzo della CPU o di altri parametri delle tue istanze Amazon EC2 e avviare automaticamente nuove istanze quando necessario. Per ulteriori informazioni, consulta la [Amazon CloudWatch User Guide](#).
- Amazon CloudWatch Logs può essere utilizzato per monitorare, archiviare e accedere ai file di log da istanze Amazon EC2 e altre CloudTrail fonti. CloudWatch I log possono monitorare le informazioni nei file di registro e avvisarti quando vengono raggiunte determinate soglie. Puoi inoltre archiviare i dati del log in storage estremamente durevole. Per ulteriori informazioni, consulta la [Amazon CloudWatch Logs User Guide](#).
- AWS CloudTrail acquisisce chiamate API ed eventi correlati da parte di o per conto del tuo Account AWS. Quindi, invia i file di log a un bucket Amazon S3 da te specificato. Puoi identificare quali utenti e account hanno richiamato AWS, l'indirizzo IP di origine da cui sono state effettuate le chiamate e quando sono avvenute. Per ulteriori informazioni, consultare la [Guida per l'utente AWS CloudTrail](#).

## Monitoraggio AWS dell'IoT FleetWise con Amazon CloudWatch

Le CloudWatch metriche di Amazon sono un modo per monitorare AWS le tue risorse e il loro rendimento. AWS L'IoT FleetWise invia metriche a CloudWatch. Puoi utilizzare il AWS Management Console, AWS CLI, o un'API per elencare le metriche a CloudWatch cui l'AWS IoT FleetWise invia. Per ulteriori informazioni, consulta la [Amazon CloudWatch User Guide](#).

### Important

È necessario configurare le impostazioni in modo che l'AWS IoT FleetWise possa inviare metriche a CloudWatch. Per ulteriori informazioni, consulta [Configurazione delle impostazioni](#).

Lo spazio dei nomi AWS/IoTFleetWise include i parametri descritti di seguito.

### Metriche dei segnali

| Metrica                | Description                                                                                                                                                                                                                                                                      |
|------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IllegalMessageFromEdge | <p>Un messaggio inviato dal veicolo e ricevuto dall'AWSIoT FleetWise non corrispondeva al formato richiesto.</p> <p>Unità: numero</p> <p>Dimensioni: VehicleName</p> <p>Statistiche valide: somma</p>                                                                            |
| MessageThrottled       | <p>Un messaggio inviato dal veicolo all'AWSIoT FleetWise è stato interrotto. Questo perché hai superato i <a href="#">limiti di servizio</a> per questo account nella regione corrente.</p> <p>Unità: numero</p> <p>Dimensioni: VehicleName</p> <p>Statistiche valide: somma</p> |
| ModelingError          | <p>Un messaggio inviato dal veicolo e ricevuto dall'AWSIoT FleetWise contiene segnali che non vengono convalidati rispetto al modello del veicolo.</p> <p>Unità: numero</p> <p>Dimensioni: ModelManifestName</p> <p>Statistiche valide: somma</p>                                |
| DecodingError          | <p>Un messaggio inviato dal veicolo e ricevuto dall'AWSIoT FleetWise contiene segnali che non vengono decodificati rispetto al manifesto del decoder del veicolo.</p>                                                                                                            |

| Metrica | Description                                                           |
|---------|-----------------------------------------------------------------------|
|         | Unità: numero<br>Dimensioni: DecoderName<br>Statistiche valide: somma |

## Parametri della campagna

| Metrica          | Description                                                                                                                                                                    |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| VehicleNotFound  | Un messaggio inviato dal veicolo e ricevuto dall'AWSIoT FleetWise, in cui il veicolo è sconosciuto.<br>Unità: numero<br>Dimensioni: VehicleName<br>Statistiche valide: somma   |
| CampaignInvalid  | Un messaggio inviato dal veicolo e ricevuto dall'AWSIoT FleetWise, in cui la campagna non è valida.<br>Unità: numero<br>Dimensioni: CampaignName<br>Statistiche valide: somma  |
| CampaignNotFound | Un messaggio inviato dal veicolo e ricevuto dall'AWSIoT FleetWise, in cui la campagna è sconosciuta.<br>Unità: numero<br>Dimensioni: CampaignName<br>Statistiche valide: somma |

## Metriche di destinazione dei dati della campagna

| Metrica              | Description                                                                                                                                                                                                                 |
|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TimestreamWriteError | <p>AWSL'IoT non è FleetWise riuscito a scrivere un messaggio dal veicolo alla tabella Amazon Timestream.</p> <p>Unità: numero</p> <p>Dimensioni: DatabaseName TableName</p> <p>Statistiche valide: somma</p>                |
| S3 WriteError        | <p>AWSL'IoT non è FleetWise riuscito a scrivere un messaggio dal veicolo al bucket Amazon Simple Storage Service (Amazon S3).</p> <p>Unità: numero</p> <p>Dimensioni: BucketName</p> <p>Statistiche valide: somma</p>       |
| S3 ReadError         | <p>AWSL'IoT non è FleetWise riuscito a leggere una chiave oggetto dal veicolo nel bucket Amazon Simple Storage Service (Amazon S3).</p> <p>Unità: numero</p> <p>Dimensioni: BucketName</p> <p>Statistiche valide: somma</p> |

## Metriche AWS KMS chiave gestite dal cliente

| Metrica             | Description                                                                                         |
|---------------------|-----------------------------------------------------------------------------------------------------|
| KMS KeyAccessDenied | <p>AWSL'IoT non è FleetWise riuscito a scrivere un messaggio dal veicolo alla tabella Timestrea</p> |

| Metrica | Description                                                                                                                                                                  |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|         | <p>mo al bucket Amazon S3 a causa di AWS KMS un errore di accesso negato alla chiave.</p> <p>Unità: numero</p> <p>Dimensioni: KMS KeyId</p> <p>Statistiche valide: somma</p> |

## Monitoraggio AWS dell'IoT FleetWise con Amazon CloudWatch Logs

Amazon CloudWatch Logs monitora gli eventi che si verificano nelle tue risorse e ti avvisa in caso di problemi. Se ricevi un avviso, puoi accedere ai file di registro per ottenere informazioni sull'evento specifico. Per ulteriori informazioni, consulta la [Amazon CloudWatch Logs User Guide](#).

### Visualizzazione dei FleetWise log AWS IoT nella console CloudWatch

#### Important

Prima di poter visualizzare il gruppo di FleetWise log AWS IoT nella CloudWatch console, assicurati che sia vero quanto segue:

- Hai abilitato l'accesso all'AWSIoT FleetWise. Per ulteriori informazioni sulla registrazione, consulta [Configurare la FleetWise registrazione AWS IoT](#).
- Esistono già voci di registro scritte dalle AWS IoT operazioni.

Per visualizzare i FleetWise log AWS IoT nella console CloudWatch

1. Apri la [CloudWatch console](#).
2. Nel riquadro di navigazione, scegli Registri, Gruppi di log.
3. Scegli il gruppo di log .
4. Scegliere Cerca gruppo di log. Vedrai un elenco completo degli eventi di registro generati per il tuo account.

5. Scegli l'icona di espansione per visualizzare un singolo stream e trovare tutti i log con un livello di ERROR registro pari a.

Puoi anche inserire una query nella casella di ricerca Filtra eventi. Ad esempio, puoi provare la seguente query:

```
{ $.logLevel = "ERROR" }
```

Per ulteriori informazioni sulla creazione di espressioni di filtro, consulta [Filter and pattern syntax](#) nella Amazon CloudWatch Logs User Guide.

### Example immissione del registro

```
{
 "accountId": "123456789012",
 "vehicleName": "test-vehicle",
 "message": "Unrecognized signal ID",
 "eventType": "MODELING_ERROR",
 "logLevel": "ERROR",
 "timestamp": 1685743214239,
 "campaignName": "test-campaign",
 "signalCatalogName": "test-catalog",
 "signalId": 10242
}
```

### tipi di eventi di segnale

| Tipo di evento | Description                                                                                                                                                                                                                                                                                                           |
|----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MODELING_ERROR | <p>Un messaggio inviato dal veicolo e ricevuto dall'AWS IoT FleetWise contiene segnali che non riescono a essere convalidati rispetto al modello del veicolo.</p> <p>Attributi: VehicleName, CampaignName, SignalID, signalCatalogName SignalValue, Min, Max, signalValueRange signalValueRange modelManifestName</p> |

| Tipo di evento               | Description                                                                                                                                                                                                                                                                            |
|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MESSAGGIO_ILLEGALE_FROM_EDGE | <p>Un messaggio inviato dal veicolo e ricevuto dall'AWSIoT FleetWise non corrispondeva al formato richiesto.</p> <p>Attributi: VehicleName, CampaignName, signalCatalogName</p>                                                                                                        |
| ERRORE_DECODIFICA            | <p>Un messaggio inviato dal veicolo e ricevuto dall'AWSIoT FleetWise contiene segnali che non vengono decodificati rispetto al manifesto del decoder del veicolo.</p> <p>Attributi: CampaignName, signalCatalogName, (opzionale) signalName decoderManifestName, (opzionale) s3URI</p> |

### Tipi di eventi della campagna

| Tipo di evento     | Description                                                                                                                                                           |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| VEHICLE_NOT_FOUND  | <p>Un messaggio inviato dal veicolo e ricevuto dall'AWSIoT FleetWise, in cui il veicolo era sconosciuto.</p> <p>Attributi: VehicleName, CampaignName</p>              |
| CAMPAIGN_NOT_FOUND | <p>Un messaggio inviato dal veicolo e ricevuto dall'AWSIoT FleetWise, in cui la campagna era sconosciuta.</p> <p>Attributi: VehicleName (opzionale), CampaignName</p> |
| CAMPAIGN_INVALID   | <p>Un messaggio inviato dal veicolo e ricevuto dall'AWSIoT FleetWise, in cui la campagna non era valida.</p>                                                          |

| Tipo di evento | Description                                      |
|----------------|--------------------------------------------------|
|                | Attributi: VehicleName (opzionale), CampaignName |

#### Dati della campagna, tipi di eventi di destinazione

| Tipo di evento         | Description                                                                                                                                                                                           |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TIMESTREAM_WRITE_ERROR | <p>AWSL'IoT non è FleetWise riuscito a scrivere un messaggio dal veicolo alla tabella Amazon Timestream.</p> <p>Attributi: VehicleName, CampaignName, timestreamDatabaseName, timestreamTableName</p> |
| S3_WRITE_ERROR         | <p>AWSL'IoT non è FleetWise riuscito a scrivere un messaggio dal veicolo al bucket Amazon Simple Storage Service (Amazon S3).</p> <p>Attributi: CampaignName, destinationName</p>                     |
| S3_READ_ERROR          | <p>AWSL'IoT non è FleetWise riuscito a leggere una chiave oggetto dal veicolo nel bucket Amazon Simple Storage Service (Amazon S3).</p> <p>Attributi: CampaignName, destinationName</p>               |

#### Tipi di eventi chiave gestiti dal cliente AWS KMS

| Tipo di evento        | Description                                                                                                                                                                            |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| KMS_KEY_ACCESS_DENIED | <p>AWSL'IoT non è FleetWise riuscito a scrivere un messaggio dal veicolo alla tabella Timestream o al bucket Amazon S3 a causa di AWS KMS un errore di accesso negato alla chiave.</p> |

## Attributes

Tutte le voci dei CloudWatch log includono i seguenti attributi:

### accountId

Il tuo ID Account AWS.

### eventType

Tipo di evento per cui il log è stato generato. Il valore del tipo di evento dipende dall'evento che ha generato la voce di registro. Ogni descrizione della voce di registro include il valore di eventType per tale voce di registro.

### logLevel

Il livello di registro utilizzato. Per ulteriori informazioni, consulta [Log levels](#) nella AWS IoT Core Developer Guide.

### message

Contiene dettagli specifici sul registro.

### timestamp

Il timestamp dell'epoca in millisecondi in cui AWS IoT ha elaborato il registro. FleetWise

## Attributi facoltativi

CloudWatch Le voci dei log includono facoltativamente questi attributi, a seconda di: eventType

### decoderManifestName

Il nome del manifesto del decodificatore che contiene il segnale.

### Nome di destinazione

Il nome della destinazione per i dati del veicolo. Ad esempio, il nome del bucket Amazon S3.

### Nome della campagna

Il nome della campagna.

### signalCatalogName

Il nome del catalogo dei segnali che contiene il segnale.

## SignalID

L'ID del segnale di errore.

## SignalID

Un elenco di ID dei segnali di errore.

## SignalName

Il nome del segnale.

## signalTimestampEpochSig.ra

Il timestamp del segnale di errore.

## SignalValue

Il valore del segnale di errore.

## signalValueRangeMax

Intervallo massimo del segnale di errore.

## signalValueRangeMin

Intervallo minimo del segnale di errore.

## S3URI

L'identificatore univoco Amazon S3 di un file Amazon Ion proveniente da un messaggio del veicolo.

## timestreamDatabaseName

Nome del database Timestream.

## timestreamTableName

Nome della tabella Timestream.

## Nome del veicolo

Il nome del veicolo.

## Configurare la FleetWise registrazione AWS IoT

È possibile inviare i dati di FleetWise log AWS IoT a un gruppo di CloudWatch log. CloudWatch I log forniscono visibilità nel caso in cui l'AWSIoT FleetWise non riesca a elaborare i messaggi dai

veicoli. Ad esempio, ciò può accadere a causa di una configurazione difettosa o di altri errori del client. Riceverai una notifica di eventuali errori in modo da poter identificare e mitigare i problemi.

Prima di poter inviare i log a CloudWatch, è necessario creare un gruppo di CloudWatch log. Configura il gruppo di log con lo stesso account e nella stessa regione che hai usato con AWS IoT FleetWise. Quando abiliti la registrazione in AWS IoT FleetWise, fornisci il nome del gruppo di log. Dopo aver abilitato la registrazione, AWS IoT FleetWise invia i log al gruppo di CloudWatch log nei flussi di log.

È possibile visualizzare i dati di registro inviati dall'AWS IoT FleetWise nella CloudWatch console. Per ulteriori informazioni sulla configurazione di un gruppo di CloudWatch log e sulla visualizzazione dei dati di log, consulta [Lavorare con i gruppi di log](#).

## Autorizzazioni per la pubblicazione dei log CloudWatch

La configurazione della registrazione per un gruppo di CloudWatch log richiede le impostazioni delle autorizzazioni descritte in questa sezione. Per informazioni sulla gestione delle autorizzazioni, consulta la sezione [Gestione degli accessi alle AWS risorse nella IAM User Guide](#).

Con queste autorizzazioni, puoi modificare la configurazione di registrazione, configurare la consegna dei log e recuperare informazioni sul tuo gruppo di log. CloudWatch

```
{
 "Version": "2012-10-17",
 "Statement": [
 {
 "Action": [
 "iotfleetwise:PutLoggingOptions",
 "iotfleetwise:GetLoggingOptions"
],
 "Resource": [
 "*"
],
 "Effect": "Allow",
 "Sid": "IoTFleetwiseLoggingOptionsAPI"
 }
],
 {
 "Sid": "IoTFleetwiseLoggingCWL",
 "Action": [
 "logs:CreateLogDelivery",

```

```
 "logs:GetLogDelivery",
 "logs:UpdateLogDelivery",
 "logs>DeleteLogDelivery",
 "logs:ListLogDeliveries",
 "logs:PutResourcePolicy",
 "logs:DescribeResourcePolicies",
 "logs:DescribeLogGroups"
],
 "Resource": [
 "*"
],
 "Effect": "Allow"
}
]
```

Quando sono consentite azioni su tutte le AWS risorse, ciò è indicato nella politica con un'"Resource" impostazione di. "\*" Ciò significa che le azioni sono consentite su tutte le AWS risorse supportate da ciascuna azione.

## Configurare la registrazione in AWS IoT FleetWise (console)

Questa sezione descrive come utilizzare la FleetWise console AWS IoT per configurare la registrazione.

Per utilizzare la FleetWise console AWS IoT per configurare la registrazione

1. Apri la [FleetWiseconsole AWS IoT](#).
2. Nel riquadro a sinistra, scegliere Settings (Impostazioni).
3. Nella sezione Registrazione della pagina Impostazioni, scegli Modifica.
4. Nella sezione CloudWatch Registrazione, inserisci il gruppo Log.
5. Per salvare le modifiche, scegli Invia.

Dopo aver abilitato la registrazione, puoi visualizzare i dati di registro nella [CloudWatch console](#).

## Configurazione della registrazione predefinita in AWS IoT FleetWise (CLI)

Questa sezione descrive come configurare la registrazione per l'AWSIoT FleetWise utilizzando la CLI.

È inoltre possibile eseguire questa procedura con l'API utilizzando i metodi dell'API AWS corrispondenti ai comandi CLI illustrati di seguito. È possibile utilizzare l'operazione [GetLoggingOptions](#) API per recuperare la configurazione corrente e l'operazione [PutLoggingOptions](#) API per modificare la configurazione.

Per utilizzare la CLI per configurare la registrazione per IoT AWS FleetWise

1. Per ottenere le opzioni di registrazione per il tuo account, usa il comando. `get-logging-options`

```
aws iotfleetwise get-logging-options
```

2. Per abilitare la registrazione, usa il `put-logging-options` comando.

```
aws iotfleetwise put-logging-options --cloud-watch-log-delivery
logType=ERROR,logGroupName=MyLogGroup
```

dove:

`logType`

Il tipo di registro per inviare i dati ai CloudWatch registri. Per disabilitare la registrazione, modifica il valore in. OFF

`logGroupName`

Il gruppo CloudWatch Logs a cui l'operazione invia i dati. Assicurati di creare il nome del gruppo di log prima di abilitare la registrazione per AWS IoT FleetWise.

Dopo aver abilitato la registrazione, consulta [Ricerca nelle voci di registro utilizzando la AWS CLI](#).

## RegistrazioneAWSIoT FleetWiseChiamate API utilizzandoAWS CloudTrail

AWSIoT FleetWiseè integrato conAWS CloudTrail, un servizio che fornisce una registrazione delle azioni intraprese da un utente, da un ruolo o da unAWS servizio inAWSIoT FleetWise. CloudTrailcattura tutte le chiamate API perAWSIoT FleetWisecome eventi. Le chiamate acquisite includono le chiamate dalAWSIoT FleetWiseconsole e chiamate in codice alAWSIoT FleetWiseOperazioni API. Se crei un percorso, puoi abilitare la consegna continua diCloudTraileventi in un bucket Amazon S3, inclusi eventi perAWSIoT FleetWise. Se non configuri

un trail, è comunque possibile visualizzare gli eventi più recenti nella console di CloudTrail in Event history (Cronologia eventi). Utilizzo delle informazioni raccolte da CloudTrail, puoi determinare la richiesta che è stata fatta a AWS IoT FleetWise, 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 su CloudTrail, consulta la [Guida per l'utente di AWS CloudTrail](#).

## AWS IoT FleetWise informazioni in CloudTrail

CloudTrail è abilitato sull'account AWS al momento della sua creazione. Quando l'attività si verifica in AWS IoT FleetWise, tale attività è registrata in un CloudTrail evento insieme ad altri AWS eventi di servizio in Cronologia degli eventi. È possibile visualizzare, cercare e scaricare gli eventi recenti nell'account AWS. Per ulteriori informazioni, consulta [Visualizzazione di eventi mediante la cronologia eventi di CloudTrail](#).

Per una registrazione continua degli eventi nel tuo AWS account, inclusi eventi per AWS IoT FleetWise, crea un sentiero. Un pista abilita CloudTrail per inviare file di registro a un bucket Amazon S3. Per impostazione predefinita, quando si crea un trail nella console, il trail sarà valido in tutte le regioni AWS. Il trail registra gli eventi di tutte le Regioni nella partizione AWS e distribuisce i file di registro nel bucket Amazon S3 specificato. Inoltre, puoi configurare altri servizi AWS per analizzare con maggiore dettaglio e usare i dati raccolti nei log CloudTrail. Per ulteriori informazioni, consulta gli argomenti seguenti:

- [Panoramica della creazione di un percorso](#)
- [Servizi e integrazioni CloudTrail supportati](#)
- [Configurazione delle notifiche Amazon SNS per CloudTrail](#)
- [Ricezione di file di log CloudTrail da più regioni](#)
- [Ricezione di file di log CloudTrail da più account](#)

Tutti AWS IoT FleetWise le azioni vengono registrate da CloudTrail e sono documentati nel [AWS IoT FleetWise Riferimento API](#). Ad esempio, le chiamate alle operazioni CreateCampaign, AssociateVehicleFleet e GetModelManifest generano voci nei file di log CloudTrail.

Ogni evento o voce di log contiene informazioni sull'utente che ha generato la richiesta. Le informazioni di identità consentono di determinare quanto segue:

- Se la richiesta è stata effettuata con le credenziali dell'utente IAM o root.

- Se la richiesta è stata effettuata con le credenziali di sicurezza temporanee per un ruolo o un utente federato.
- Se la richiesta è stata effettuata da un altro servizio AWS.

Per ulteriori informazioni, consulta [Elemento CloudTrail userIdentity](#).

## ComprensioneAWSIoT FleetWisevoci dei file di registro

Un percorso è una configurazione che consente la distribuzione di eventi come i file di log in un bucket Amazon S3 specificato. I file di log di CloudTrail contengono una o più voci di log. Un evento rappresenta una singola richiesta da un'fonte e include informazioni sul operazione richiesta, data e ora dell'operazione, parametri richiesti e così via. I file di log di CloudTrail non sono una traccia stack ordinata delle chiamate pubbliche dell'API, quindi non vengono visualizzati in un ordine specifico.

L'esempio seguente mostra una voce di log di CloudTrail che illustra l'operazione

*AssociateVehicleFleet*:

```
{
 "eventVersion": "1.05",
 "userIdentity": {
 "type": "AssumedRole",
 "principalId": "AIDACKCEVSQ6C2EXAMPLE",
 "arn": "arn:aws:iam::111122223333:assumed-role/NikkiWolf",
 "accountId": "111122223333",
 "accessKeyId": "access-key-id",
 "userName": "NikkiWolf"
 },
 "eventTime": "2021-11-30T09:56:35Z",
 "eventSource": "iotfleetwise.amazonaws.com",
 "eventName": "AssociateVehicleFleet",
 "awsRegion": "us-east-1",
 "sourceIPAddress": "192.0.2.21",
 "userAgent": "aws-cli/2.3.2 Python/3.8.8 Darwin/18.7.0 botocore/2.0.0",
 "requestParameters": {
 "fleetId": "f1234567890",
 "vehicleId": "v0213456789"
 },
 "responseElements": {
 },
 "requestID": "9f861429-11e3-11e8-9eea-0781b5c0ac21",
 "eventID": "17385819-4927-41ee-a6a5-29ml0br812v4",
```

```
"eventType": "AwsApiCall",
"recipientAccountId": "111122223333"
}
```

# Cronologia dei documenti per la AWS IoT FleetWise Developer Guide

La tabella seguente descrive le versioni della documentazione per l'AWS IoT FleetWise.

| Modifica                                                  | Descrizione                                                                                                                                                                                                                                                                                                                                                                                        | Data             |
|-----------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
| <a href="#">Anteprima dei dati del sistema di visione</a> | Puoi utilizzare l'anteprima dei dati dei sistemi di visione dell'AWS IoT FleetWise per raccogliere e organizzare i dati dai sistemi di visione dei veicoli, tra cui telecamere, radar e lidar. Mantiene automaticamente sincronizzati nel cloud i dati del sistema di visione strutturati e non strutturati, i metadati (ID evento, campagna, veicolo) e il sensore standard (dati di telemetria). | 26 novembre 2023 |
| <a href="#">AWS KMSchiavi gestite dal cliente</a>         | AWSL'IoT FleetWise ora supporta le chiavi gestite dai AWS KMS clienti. Puoi utilizzare la chiave KMS per crittografare i dati lato server relativi alle FleetWise risorse AWS IoT (catalogo dei segnali, modello del veicolo, manifesto del decoder, veicoli e configurazioni della campagna di raccolta dati) archiviate in. Cloud AWS                                                            | 16 ottobre 2023  |

[Archiviazione di oggetti in Amazon S3](#)

AWS IoT FleetWise ora supporta l'archiviazione dei dati tramite Amazon Simple Storage Service (Amazon S3). Puoi archiviare i dati raccolti durante le campagne in Amazon S3, oltre ad Amazon Timestream.

1 giugno 2023

[Disponibilità generale](#)

Questa è la versione pubblica dell'AWS IoT FleetWise.

27 settembre 2022

[Versione iniziale](#)

Questa è la versione di anteprima della AWS IoT FleetWise Developer Guide.

30 novembre 2021

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.