AWS Ground Station Guida per l'utente dell'agente

AWS Ground Station

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Ground Station: AWS Ground Station Guida per l'utente dell'agente

Copyright © 2025 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

Panoramica ... 1
Cos'è l' AWS Ground Station agente? ... 1
Caratteristiche dell'agente AWS Ground Station ... 2
Requisiti degli agenti ... 3
Diagrammi VPC ... 4
Sistema operativo supportato ... 5
Ricevi dati tramite l'agente AWS Ground Station ... 6
Flussi di dati multipli, ricevitore singolo ... 6
Flussi di dati multipli, ricevitori multipli ... 7
Seleziona l' EC2 istanza Amazon e prenota i core della CPU per la tua architettura ... 9
Tipi di EC2 istanze Amazon supportati ... 9
Pianificazione del core della CPU ... 10
Raccolta di informazioni sull'architettura ... 11
Esempio di assegnazione della CPU ... 13
Appendice: `lscpu -p` output (completo) per c5.24xlarge ... 13
Installare l'agente ... 17
Usa un AWS CloudFormation modello ... 17
Fase 1: Creare risorse AWS ... 17
Fase 2: Verifica lo stato dell'agente ... 17
Installa manualmente su EC2 ... 17
Fase 1: Creare risorse AWS ... 17
Fase 2: Creare un'istanza EC2 ... 18
Passaggio 3: scarica e installa l'agente ... 18
Fase 4: Configurare l'agente ... 19
Fase 5: Applicare l'ottimizzazione delle prestazioni ... 20
Fase 6: Gestire l'agente ... 20
Gestisci l'agente ... 21
AWS Ground Station Configurazione dell'agente ... 21
AWS Ground Station Inizio dell'agente ... 21
AWS Ground Station Agente, fermati ... 22
AWS Ground Station Aggiornamento dell'agente ... 22
AWS Ground Station Downgrade dell'agente ... 23
AWS Ground Station Disinstallazione dell'agente ... 24
AWS Ground Station Stato dell'agente ... 24

AWS Ground Station Informazioni sull'RPM dell'agente ... 25
Configura l'agente ... 26
File di configurazione dell'agente ... 26
Esempio ... 26
Suddivisione del campo ... 26
Ottimizza la tua EC2 istanza per migliorarne le prestazioni ... 30
Ottimizza le interruzioni hardware e le code di ricezione: influisce sulla CPU e sulla rete ... 30
Tune Rx interrupt coalescenza: influisce sulla rete ... 31
Tune Rx ring buffer: influisce sulla rete ... 32
Tune CPU C-State: influisce sulla CPU ... 32
Riserva le porte di ingresso: influisce sulla rete ... 32
Riavvio ... 33
Appendice: Parametri consigliati per l'accordatura interrupt/RPS ... 33
Preparati a prendere un contatto DigiF ... 36
Best practice ... 37
EC2 Le migliori pratiche di Amazon ... 37
Pianificatore Linux ... 37
AWS Ground Station elenco di prefissi gestiti ... 37
Limitazione del contatto singolo ... 37
Esecuzione di servizi e processi insieme all'agente AWS Ground Station ... 37
Ad esempio, utilizzando un'`c5.24xlarge`istanza ... 38
Servizi di affinitizzazione (systemd) ... 38
Affinitizzazione dei processi (script) ... 39
Risoluzione dei problemi ... 41
L'agente non riesce ad avviarsi ... 41
Risoluzione dei problemi ... 41
AWS Ground Station Registri degli agenti ... 42
Nessun contatto disponibile ... 42
Ottenere supporto ... 43
Note sulla versione dell'agente ... 44
Versione più recente dell'agente ... 44
Versione 1.0.3555.0 ... 44
Versioni obsolete degli agenti ... 44
Versione 1.0.2942.0 ... 44
Versione 1.0.2716.0 ... 45
Versione 1.0.2677.0 ... 45

Convalida dell'installazione RPM 47
Versione più recente dell'agente 44
Versione 1.0.3555.0 44
Verifica l'RPM 48
Cronologia dei documenti 49
..... I

Panoramica

Cos'è l' AWS Ground Station agente?

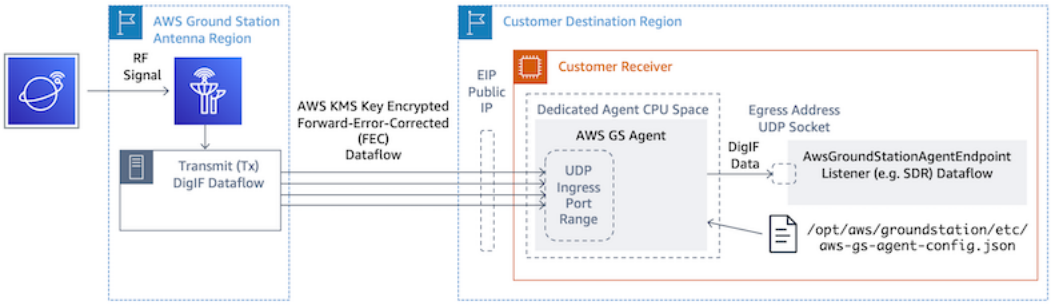
Con l' AWS Ground Station agente, disponibile come RPM, puoi ricevere (downlink) flussi di dati sincroni Wideband Digital Intermediate Frequency (DigiF) durante i contatti con AWS Ground Station. Puoi selezionare due opzioni per la consegna dei dati:

1. Consegna dei dati a un' EC2 istanza: consegna dei dati a un' EC2 istanza di tua proprietà. Sei tu a gestire l' AWS Ground Station agente. Questa opzione può essere la soluzione migliore se è necessaria un'elaborazione dei dati quasi in tempo reale. Consulta la guida Data Delivery to Amazon Elastic Compute Cloud per informazioni sulla distribuzione EC2 dei dati.
2. Distribuzione dei dati a un bucket S3: consegna dei dati a un bucket AWS S3 di tua proprietà tramite un servizio gestito Ground Station. Consulta la AWS Ground Station guida introduttiva per informazioni sulla distribuzione dei dati S3.

Entrambe le modalità di distribuzione dei dati richiedono la creazione di un set di risorse AWS. L'uso di CloudFormation per creare le tue risorse AWS è altamente consigliato per garantire affidabilità, precisione e supportabilità. Ogni contatto può fornire dati solo a EC2 o a S3 ma non a entrambi contemporaneamente.

> **Note**
>
> Poiché la distribuzione dei dati di S3 è un servizio gestito da Ground Station, questa guida si concentra sulla consegna dei dati alla/e EC2 istanza/e.

Il diagramma seguente mostra un flusso di dati DigiF da AWS Ground Station una regione di antenna all'istanza con EC2 il Software-Defined Radio (SDR) o un ascoltatore simile.



Caratteristiche dell'agente AWS Ground Station

L' AWS Ground Station agente riceve dati di downlink Digital Intermediate Frequency (DigiF) e trasmette dati decrittografati che consentono quanto segue:

- Capacità di downlink DigiF da 40 MHz a 400 MHz di larghezza di banda.
- Distribuzione di dati DigiF ad alta velocità e basso jitter a qualsiasi IP pubblico (AWS Elastic IP) sulla rete AWS.
- Distribuzione affidabile dei dati tramite Forward Error Correction (FEC).
- Distribuzione sicura dei dati utilizzando una AWS KMS chiave di crittografia gestita dal cliente.

Requisiti degli agenti

> **Note**
>
> Questa guida per AWS Ground Station agenti presuppone che tu abbia effettuato l'onboarding a Ground Station utilizzando la guida AWS Ground Station introduttiva.

L' EC2 istanza del ricevitore AWS Ground Station Agent richiede un set di risorse AWS dipendenti per fornire in modo affidabile e sicuro i dati DigiF ai tuoi endpoint.

1. Un VPC in cui avviare il EC2 ricevitore.
2. Una chiave AWS KMS per la crittografia/decrittografia dei dati.
3. Una chiave SSH o un profilo di EC2 istanza configurato per SSM Session Manager.
4. Regole del gruppo di rete/sicurezza per consentire quanto segue:
    1. Traffico UDP proveniente dalle AWS Ground Station porte specificate nel gruppo di endpoint dataflow. L'agente riserva una serie di porte contigue utilizzate per fornire dati agli endpoint del flusso di dati in ingresso.
    2. Accesso SSH alla tua istanza (Nota: in alternativa puoi usare AWS Session Manager per accedere alla tua EC2 istanza).
    3. Accesso in lettura a un bucket S3 accessibile pubblicamente per la gestione degli agenti.
    4. Traffico SSL sulla porta 443 che consente all'agente di comunicare con il servizio. AWS Ground Station
    5. Traffico proveniente dall'elenco dei AWS Ground Station prefissi gestiti. `com.amazonaws.global.groundstation`

Inoltre, è richiesta una configurazione VPC che includa una sottorete pubblica. Consulta la Guida utente VPC per informazioni di base sulla configurazione della sottorete.

Configurazioni compatibili:

1. Un IP elastico associato all' EC2 istanza in una sottorete pubblica.
2. Un IP elastico associato a un ENI in una sottorete pubblica, collegato all' EC2 istanza (in qualsiasi sottorete nella stessa zona di disponibilità della sottorete pubblica).

Puoi utilizzare lo stesso gruppo di sicurezza dell' EC2 istanza o specificarne uno con almeno il set minimo di regole composto da:

- Traffico UDP proveniente AWS Ground Station dalle porte specificate nel gruppo di endpoint dataflow.

Ad esempio, modelli di AWS CloudFormation EC2 Data Delivery con queste risorse preconfigurate, vedi Public broadcast satellite using AWS Ground Station Agent (wideband).

Diagrammi VPC

Diagramma: un IP elastico associato all' EC2 istanza in una sottorete pubblica

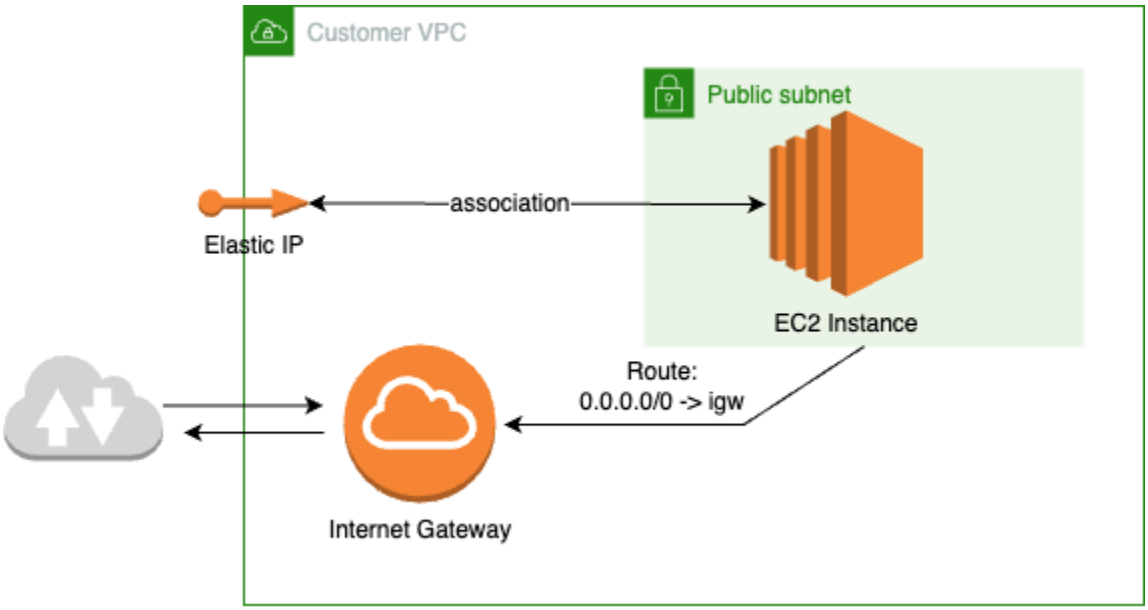
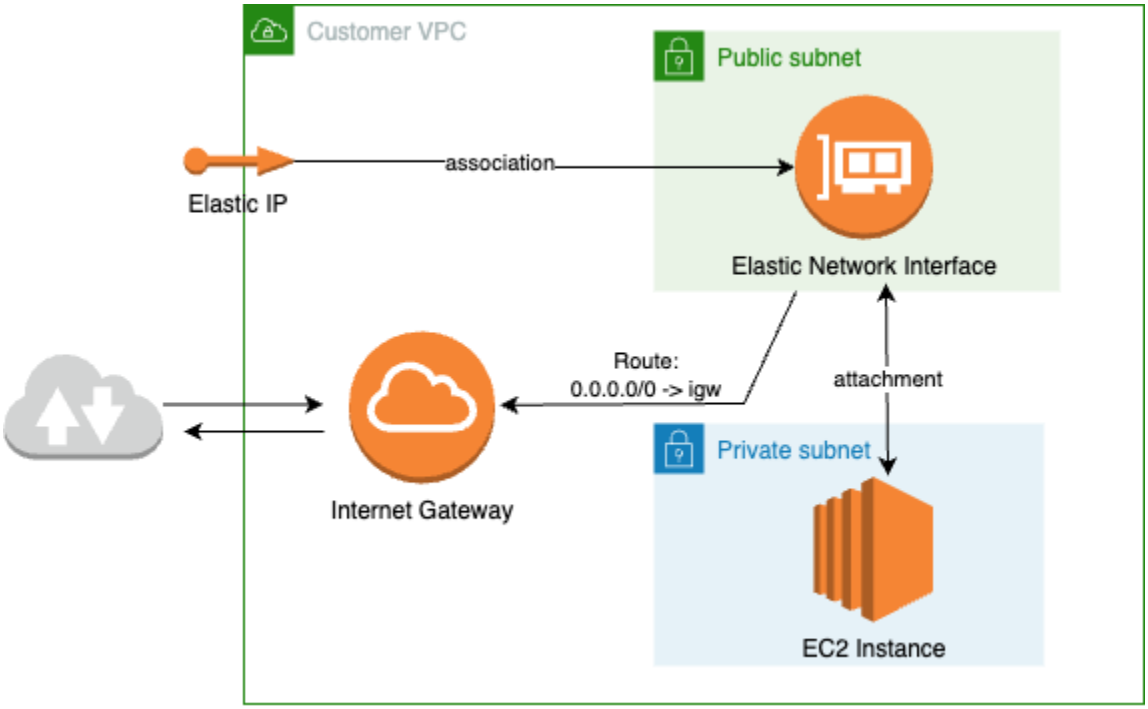


Diagramma: un IP elastico associato a un ENI in una sottorete pubblica, collegato all' EC2 istanza in una sottorete privata



Sistema operativo supportato

Amazon Linux 2 con kernel 5.10+.

I tipi di istanze supportati sono elencati in Seleziona l' EC2 istanza Amazon e prenota i core della CPU per la tua architettura

Ricevi dati tramite l'agente AWS Ground Station

I diagrammi seguenti forniscono una panoramica del flusso di dati AWS Ground Station durante i contatti DiGIF (Wideband Digital Intermediate Frequency).

L' AWS Ground Station agente gestirà l'orchestrazione dei componenti del dataplane per un contatto. Prima di pianificare un contatto, l'agente deve essere configurato e avviato correttamente e deve essere registrato (la registrazione è automatica all'avvio dell'agente) con. AWS Ground Station Inoltre, il software di ricezione dei dati (ad esempio una radio definita dal software) deve essere in esecuzione e configurato per ricevere dati all'indirizzo AwsGroundStationAgentEndpointEgressAddress.

Dietro le quinte, l' AWS Ground Station agente riceverà le operazioni AWS Ground Station e annullerà la AWS KMS crittografia applicata in transito, prima di inoltrarle all'endpoint EgressAddress di destinazione dove è in ascolto la Software Defined Radio (SDR). L' AWS Ground Station agente e i suoi componenti sottostanti rispetteranno i limiti della CPU impostati nel file di configurazione per garantire che non influiscano sulle prestazioni di altre applicazioni in esecuzione sull'istanza.

È necessario che l' AWS Ground Station agente sia in esecuzione sull'istanza ricevente coinvolta nel contatto. Un singolo AWS Ground Station agente è in grado di orchestrare più flussi di dati, come illustrato di seguito, se si preferisce ricevere tutti i flussi di dati su un'unica istanza del ricevitore.

Flussi di dati multipli, ricevitore singolo

Scenario di esempio:

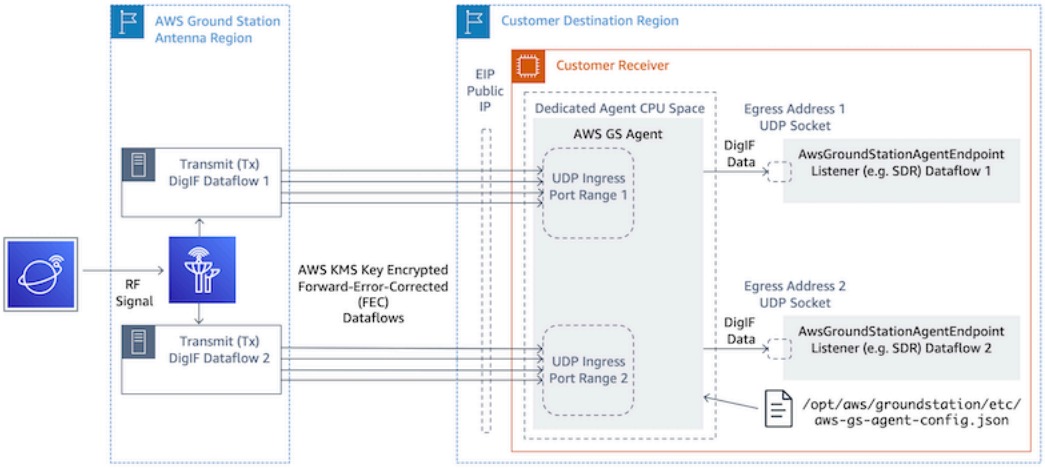
Desideri ricevere due downlink di antenne come flussi di dati DigiF sulla stessa istanza del ricevitore. EC2 I due downlink saranno 200 e 100. MHz MHz

AwsGroundStationAgentEndpoints:

Saranno disponibili due `AwsGroundStationAgentEndpoint` risorse, una per ogni flusso di dati. Entrambi gli endpoint avranno lo stesso indirizzo IP pubblico (). `ingressAddress.socketAddress.name` Gli ingressi `portRange` non devono sovrapporsi, poiché i flussi di dati vengono ricevuti nella stessa istanza. EC2 Entrambi devono essere `egressAddress.socketAddress.port` unici.

Pianificazione della CPU:

- 1 core (2 vCPU) per l'esecuzione del singolo AWS Ground Station agente sull'istanza.
- 6 core (12 vCPU) per ricevere DigiF Dataflow 1 (200 ricerche nella tabella). MHz Pianificazione del core della CPU
- 4 core (8 vCPU) per ricevere DigiF Dataflow 2 (100 ricerche nella tabella). MHz Pianificazione del core della CPU
- Spazio totale dedicato sulla CPU dell'agente = 11 core (22 vCPU) sullo stesso socket.



Flussi di dati multipli, ricevitori multipli

Scenario di esempio:

Desideri ricevere due downlink di antenne come flussi di dati DigiF su diverse istanze del ricevitore. EC2 Entrambi i downlink saranno 400. MHz

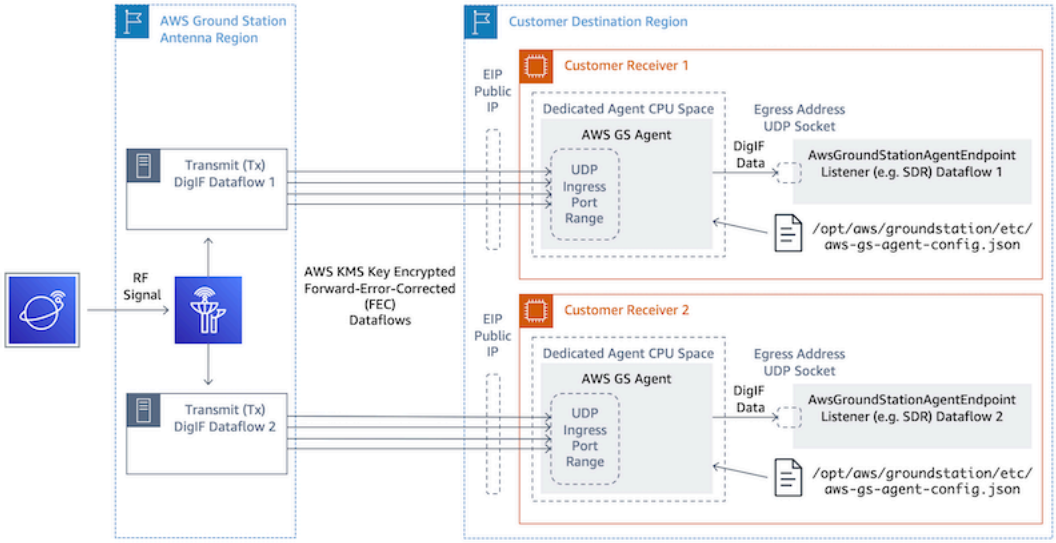
AwsGroundStationAgentEndpoints:

Saranno disponibili due `AwsGroundStationAgentEndpoint` risorse, una per ogni flusso di dati. Gli endpoint avranno un indirizzo IP pubblico diverso (). `ingressAddress.socketAddress.name` Non vi sono restrizioni sui valori delle porte per nessuna delle due `ingressAddress` o `egressAddress` poiché i flussi di dati vengono ricevuti su un'infrastruttura separata e non sono in conflitto tra loro.

Pianificazione della CPU:

- Istanza del ricevitore 1

  - 1 core (2 vCPU) per l'esecuzione del singolo AWS Ground Station agente sull'istanza.
  - 9 core (18 vCPU) per ricevere DigiF Dataflow 1 (ricerca 400 nella tabella). MHz Pianificazione del core della CPU
  - Spazio CPU agente dedicato totale = 10 core (20 vCPU) sullo stesso socket.
- Istanza del ricevitore 2
  - 1 core (2 vCPU) per l'esecuzione del singolo AWS Ground Station agente sull'istanza.
  - 9 core (18 vCPU) per ricevere DigiF Dataflow 2 (ricerca 400 nella tabella). MHz Pianificazione del core della CPU
  - Spazio CPU agente dedicato totale = 10 core (20 vCPU) sullo stesso socket.



Seleziona l' EC2 istanza Amazon e prenota i core della CPU per la tua architettura

Tipi di EC2 istanze Amazon supportati

L' AWS Ground Station Agent richiede core CPU dedicati per funzionare a causa dei flussi di lavoro di distribuzione dei dati ad alta intensità di calcolo. Supportiamo i seguenti tipi di istanze. Vedi Pianificazione del core della CPU per decidere quale tipo di istanza si adatta meglio al tuo caso d'uso.

Famiglia di istanze	Tipo di istanza	Impostazione predefinita v CPUs	Core CPU predefiniti
c5	c5.12xlarge	48	24
	c5.18xlarge	72	36
	c5.24xlarge	96	48
c5n	c5n.18xlarge	72	36
	c5n.metal	72	36
c6i	c6i.24xlarge	96	48
	c6i.32xlarge	128	64
p3dn	p3dn.24xlarge	96	48
g4dn	g4dn.12xlarge	48	24
	g4dn.16xlarge	64	32
	g4dn.metal	96	48
p4d	p4d.24xlarge	96	48
m5	m5.8xlarge	32	16

Famiglia di istanze	Tipo di istanza	Impostazione predefinita v CPUs	Core CPU predefiniti
	m5.12xlarge	48	24
	m5.24xlarge	96	48
m6i	m6i.32xlarge	128	64
r5	r5.24xlarge	96	48
	r5.metal	96	48
r5n	r5n.24xlarge	96	48
	r5n.metal	96	48
r6i	r6i.32xlarge	128	64

Pianificazione del core della CPU

L' AWS Ground Station agente richiede core di processore dedicati che condividano la cache L3 per ogni flusso di dati. L'agente è progettato per sfruttare le coppie di CPU Hyper-Threaded (HT) e richiede che le coppie HT siano riservate per il suo utilizzo. Una coppia hyper-threaded è una coppia di elementi virtuali (CPUs vCPU) contenuti in un singolo core. La tabella seguente fornisce una mappatura della velocità dei dati del flusso di dati al numero richiesto di core riservati all'agente per un singolo flusso di dati. Questa tabella presuppone Cascade Lake o versione successiva ed è valida per qualsiasi tipo di istanza supportato CPUs . Se la larghezza di banda è compresa tra le voci della tabella, seleziona la successiva più alta.

L'agente necessita di un core riservato aggiuntivo per la gestione e il coordinamento, quindi il totale dei core richiesti sarà la somma dei core necessari (dal grafico seguente) per ogni flusso di dati più un singolo core aggiuntivo (2 v). CPUs

AntennaDownlink Larghezza di banda () MHz	Velocità dati VITA-49.2 DigiF prevista (MB/s)	Numero di core (coppie di CPU HT)	vCPU totale
50	1000	3	6
100	2000	4	8
150	3000	5	10
200	4000	6	12
250	5000	6	12
300	6000	7	14
350	7000	8	16
400	8000	9	18

Raccolta di informazioni sull'architettura

`lscpu`fornisce informazioni sull'architettura del sistema. L'output di base mostra quali v CPUs (etichettati come «CPU») appartengono a quali nodi NUMA (e ogni nodo NUMA condivide una cache L3). Di seguito esaminiamo un'`c5.24xlarge`istanza per raccogliere le informazioni necessarie per configurare l'agente. AWS Ground Station Ciò include informazioni utili come il numero di vCPUs, i core e vCPU-to-node l'associazione.

```
> lscpu
Architecture: x86_64
CPU op-mode(s): 32-bit, 64-bit
Byte Order: Little Endian
CPU(s): 96
On-line CPU(s) list: 0-95
Thread(s) per core: 2             <------
Core(s) per socket: 24
Socket(s): 2
```

```
NUMA node(s): 2
Vendor ID: GenuineIntel
CPU family: 6
Model: 85
Model name: Intel(R) Xeon(R) Platinum 8275CL CPU @ 3.00GHz
Stepping: 7
CPU MHz: 3601.704
BogoMIPS: 6000.01
Hypervisor vendor: KVM
Virtualization type: full
L1d cache: 32K
L1i cache: 32K
L2 cache: 1024K
L3 cache: 36608K
NUMA node0 CPU(s): 0-23,48-71     <------
NUMA node1 CPU(s): 24-47,72-95    <------
```

I core dedicati all' AWS Ground Station agente devono includere sia v CPUs per ogni core assegnato. Tutti i core di un flusso di dati devono esistere sullo stesso nodo NUMA. L'`-`popzione per il `lscpu` comando ci fornisce le associazioni tra core e CPU necessarie per configurare l'agente. I campi pertinenti sono CPU (che è ciò che chiamiamo vCPU), Core e L3 (che indica quale cache L3 è condivisa da quel core). Si noti che sulla maggior parte dei processori Intel il nodo NUMA è uguale alla cache L3.

Considerate il seguente sottoinsieme dell'`lscpu -p`output per a `c5.24xlarge` (abbreviato e formattato per maggiore chiarezza).

```
CPU,Core,Socket,Node,,L1d,L1i,L2,L3
0   0    0      0     0   0   0  0
1   1    0      0     1   1   1  0
2   2    0      0     2   2   2  0
3   3    0      0     3   3   3  0
...
16  0    0      0     0   0   0  0
17  1    0      0     1   1   1  0
18  2    0      0     2   2   2  0
19  3    0      0     3   3   3  0
```

Dall'output possiamo vedere che Core 0 include v CPUs 0 e 16, Core 1 include v CPUs 1 e 17, Core 2 include v CPUs 2 e 18. In altre parole, le coppie hyper-threaded sono: 0 e 16, 1 e 17, 2 e 18.

Esempio di assegnazione della CPU

Ad esempio, useremo un'`c5.24xlarge`istanza per un downlink a banda larga a doppia polarità a 350. MHz Dalla tabella in basso Pianificazione del core della CPU sappiamo che un MHz downlink 350 richiede 8 core (16 vCPUs) per il singolo flusso di dati. Ciò significa che questa configurazione a doppia polarità che utilizza due flussi di dati richiede un totale di 16 core (32 vCPUs) più un core (2 v) per l'agente. CPUs

Conosciamo l'`lscpu`output per include e. `c5.24xlarge` `NUMA node0 CPU(s): 0-23,48-71` `NUMA node1 CPU(s): 24-47,72-95` Poiché NUMA node0 ha più del necessario, assegneremo solo due core: 0-23 e 48-71.

Innanzitutto, selezioneremo 8 core per ogni flusso di dati che condividono una cache L3 o un nodo NUMA. Quindi cercheremo la v corrispondente CPUs (denominata «CPU») nell'output in. `lscpu -p` Appendice: `lscpu -p` output (completo) per c5.24xlarge Un esempio di processo di selezione principale potrebbe essere simile al seguente:

- Riserva i core 0-1 per il sistema operativo.
- Flusso 1: seleziona i core 2-9 che corrispondono a v CPUs 2-9 e 50-57.
- Flusso 2: seleziona i core 10-17 che corrispondono a v 10-17 e 58-65. CPUs
- Agent core: seleziona il core 18 che corrisponde alle v 18 e 66. CPUs

Ciò si traduce in v CPUs 2-18 e 50-66, quindi l'elenco da fornire all'agente è. `[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66]` È necessario assicurarsi che i propri processi non vengano eseguiti su di essi CPUs come descritto in. Esecuzione di servizi e processi insieme all'agente AWS Ground Station

Notate che i core specifici selezionati in questo esempio sono alquanto arbitrari. Altri set di core funzionerebbero purché soddisfino il requisito di condividere tutti una cache L3 per ogni flusso di dati.

Appendice: `lscpu -p` output (completo) per c5.24xlarge

```
> lscpu -p
# The following is the parsable format, which can be fed to other
# programs. Each different item in every column has an unique ID
# starting from zero.
# CPU,Core,Socket,Node,,L1d,L1i,L2,L3
0,0,0,0,,0,0,0,0
1,1,0,0,,1,1,1,0
2,2,0,0,,2,2,2,0
3,3,0,0,,3,3,3,0
4,4,0,0,,4,4,4,0
5,5,0,0,,5,5,5,0
6,6,0,0,,6,6,6,0
7,7,0,0,,7,7,7,0
8,8,0,0,,8,8,8,0
9,9,0,0,,9,9,9,0
10,10,0,0,,10,10,10,0
11,11,0,0,,11,11,11,0
12,12,0,0,,12,12,12,0
13,13,0,0,,13,13,13,0
14,14,0,0,,14,14,14,0
15,15,0,0,,15,15,15,0
16,16,0,0,,16,16,16,0
17,17,0,0,,17,17,17,0
18,18,0,0,,18,18,18,0
19,19,0,0,,19,19,19,0
20,20,0,0,,20,20,20,0
21,21,0,0,,21,21,21,0
22,22,0,0,,22,22,22,0
23,23,0,0,,23,23,23,0
24,24,1,1,,24,24,24,1
25,25,1,1,,25,25,25,1
26,26,1,1,,26,26,26,1
27,27,1,1,,27,27,27,1
28,28,1,1,,28,28,28,1
29,29,1,1,,29,29,29,1
30,30,1,1,,30,30,30,1
31,31,1,1,,31,31,31,1
32,32,1,1,,32,32,32,1
33,33,1,1,,33,33,33,1
34,34,1,1,,34,34,34,1
35,35,1,1,,35,35,35,1
36,36,1,1,,36,36,36,1
37,37,1,1,,37,37,37,1
38,38,1,1,,38,38,38,1
```

```
39,39,1,1,,39,39,39,1
40,40,1,1,,40,40,40,1
41,41,1,1,,41,41,41,1
42,42,1,1,,42,42,42,1
43,43,1,1,,43,43,43,1
44,44,1,1,,44,44,44,1
45,45,1,1,,45,45,45,1
46,46,1,1,,46,46,46,1
47,47,1,1,,47,47,47,1
48,0,0,0,,0,0,0,0
49,1,0,0,,1,1,1,0
50,2,0,0,,2,2,2,0
51,3,0,0,,3,3,3,0
52,4,0,0,,4,4,4,0
53,5,0,0,,5,5,5,0
54,6,0,0,,6,6,6,0
55,7,0,0,,7,7,7,0
56,8,0,0,,8,8,8,0
57,9,0,0,,9,9,9,0
58,10,0,0,,10,10,10,0
59,11,0,0,,11,11,11,0
60,12,0,0,,12,12,12,0
61,13,0,0,,13,13,13,0
62,14,0,0,,14,14,14,0
63,15,0,0,,15,15,15,0
64,16,0,0,,16,16,16,0
65,17,0,0,,17,17,17,0
66,18,0,0,,18,18,18,0
67,19,0,0,,19,19,19,0
68,20,0,0,,20,20,20,0
69,21,0,0,,21,21,21,0
70,22,0,0,,22,22,22,0
71,23,0,0,,23,23,23,0
72,24,1,1,,24,24,24,1
73,25,1,1,,25,25,25,1
74,26,1,1,,26,26,26,1
75,27,1,1,,27,27,27,1
76,28,1,1,,28,28,28,1
77,29,1,1,,29,29,29,1
78,30,1,1,,30,30,30,1
79,31,1,1,,31,31,31,1
80,32,1,1,,32,32,32,1
81,33,1,1,,33,33,33,1
82,34,1,1,,34,34,34,1
```

```
83,35,1,1,,35,35,35,1
84,36,1,1,,36,36,36,1
85,37,1,1,,37,37,37,1
86,38,1,1,,38,38,38,1
87,39,1,1,,39,39,39,1
88,40,1,1,,40,40,40,1
89,41,1,1,,41,41,41,1
90,42,1,1,,42,42,42,1
91,43,1,1,,43,43,43,1
92,44,1,1,,44,44,44,1
93,45,1,1,,45,45,45,1
94,46,1,1,,46,46,46,1
95,47,1,1,,47,47,47,1
```

Installare l'agente

L' AWS Ground Station agente può essere installato nei seguenti modi:

1. AWS CloudFormation modello (consigliato).
2. Installazione manuale su Amazon EC2.

Usa un AWS CloudFormation modello

Il AWS CloudFormation modello di distribuzione EC2 dei dati crea le risorse AWS necessarie per fornire dati alla tua EC2 istanza. Questo AWS CloudFormation modello utilizza l'AMI AWS Ground Station gestita su cui è preinstallato l' AWS Ground Station agente. Lo script di avvio dell' EC2 istanza creata popola quindi il file di configurazione dell'agente e applica il necessario tuning delle prestazioni (). Ottimizza la tua EC2 istanza per migliorarne le prestazioni

Fase 1: Creare risorse AWS

Crea il tuo stack di risorse AWS utilizzando il modello Public broadcast satellite utilizzando AWS Ground Station Agent (banda larga).

Fase 2: Verifica lo stato dell'agente

Per impostazione predefinita, l'agente è configurato e attivo (avviato). Per verificare lo stato dell'agente puoi connetterti all' EC2 istanza (SSH o SSM Session Manager) e vedere. AWS Ground Station Stato dell'agente

Installa manualmente su EC2

Sebbene Ground Station consigli l'uso CloudFormation di modelli per il provisioning delle risorse AWS, potrebbero esserci casi d'uso in cui il modello standard potrebbe non essere sufficiente. In questi casi, ti consigliamo di personalizzare il modello in base alle tue esigenze. Se ciò non soddisfa ancora i tuoi requisiti, puoi creare manualmente le tue risorse AWS e installare l'agente.

Fase 1: Creare risorse AWS

Consulta Esempi di configurazioni del profilo di missione per istruzioni su come configurare manualmente le risorse AWS richieste per un contatto.

La AwsGroundStationAgentEndpointrisorsa definisce un endpoint per la ricezione di un flusso di dati DigiF AWS Ground Station tramite Agent ed è fondamentale per stabilire un contatto di successo. Sebbene la documentazione dell'API sia disponibile nell'API Reference, questa sezione discuterà brevemente i concetti relativi all'agente. AWS Ground Station

L'endpoint `ingressAddress` è il luogo in cui l' AWS Ground Station agente riceverà il traffico UDP AWS KMS crittografato dall'antenna. `socketAddressname`è l'IP pubblico dell' EC2 istanza (dall'EIP allegato). `portRange`Dovrebbero esserci almeno 300 porte contigue in un intervallo riservato a qualsiasi altro utilizzo. Per istruzioni, consulta Riserva le porte di ingresso: influisce sulla rete. Queste porte devono essere configurate per consentire il traffico di ingresso UDP sul gruppo di sicurezza per il VPC in cui è in esecuzione l'istanza del ricevitore.

L'endpoint `egressAddress` è il luogo in cui l'agente ti consegnerà il flusso di dati DigiF. Dovresti avere un'applicazione (ad esempio SDR) che riceve i dati tramite un socket UDP in questa posizione.

Fase 2: Creare un'istanza EC2

Sono AMIs supportati:

1. AWS Ground Station L'AMI, `groundstation-al2-gs-agent-ami-*` dove* è la data di creazione dell'AMI, viene fornita con l'agente installato (scelta consigliata).
2. `amzn2-ami-kernel-5.10-hvm-x86_64-gp2`.

Passaggio 3: scarica e installa l'agente

> **Note**
>
> I passaggi di questa sezione devono essere completati se non hai scelto l' AWS Ground Station Agent AMI nel passaggio precedente.

Scarica l'agente

L' AWS Ground Station agente è disponibile da bucket S3 specifici della regione e può essere scaricato su EC2 istanze di supporto utilizzando la riga di comando AWS (CLI) da `s3://groundstation-wb-digif-software-${AWS::Region}/aws-groundstation-agent/latest/amazon_linux_2_x86_64/aws-groundstation-agent.rpm` cui $ {AWS::Region} si riferisce a una delle regioni di distribuzione dati e console AWS Ground Station supportate.

Esempio: scarica l'ultima versione rpm dalla regione AWS us-east-2 localmente nella cartella /tmp.

```
aws s3 --region us-east-2 cp s3://groundstation-wb-digif-software-us-east-2/aws-
groundstation-agent/latest/amazon_linux_2_x86_64/aws-groundstation-agent.rpm /tmp
```

Se devi scaricare una versione specifica dell' AWS Ground Station agente, puoi scaricarla dalla cartella specifica della versione nel bucket S3.

Esempio: scarica la versione 1.0.2716.0 del file rpm dalla regione AWS us-east-2 localmente nella cartella /tmp.

```
aws s3 --region us-east-2 cp s3://groundstation-wb-digif-software-us-east-2/aws-
groundstation-agent/1.0.2716.0/amazon_linux_2_x86_64/aws-groundstation-agent.rpm /tmp
```

> **Note**
>
> Se vuoi confermare che l'RPM che hai scaricato è stato fornito da, segui le istruzioni per. AWS Ground StationConvalida dell'installazione RPM

Installa l'agente

```
sudo yum install ${MY_RPM_FILE_PATH}

Example: Assumes agent is in the "/tmp" directory
sudo yum install /tmp/aws-groundstation-agent.rpm
```

Fase 4: Configurare l'agente

Dopo aver installato l'agente, è necessario aggiornare il file di configurazione dell'agente. Consultare Configura l'agente.

Fase 5: Applicare l'ottimizzazione delle prestazioni

AWS Ground Station Agent AMI: se hai scelto l' AWS Ground Station Agent AMI nel passaggio precedente, applica le seguenti ottimizzazioni delle prestazioni.

- Ottimizza le interruzioni hardware e le code di ricezione: influisce sulla CPU e sulla rete
- Riserva le porte di ingresso: influisce sulla rete
- Riavvio

Altro AMIs: se hai scelto un'altra AMI nel passaggio precedente, applica tutte le regolazioni elencate sotto Ottimizza la tua EC2 istanza per migliorarne le prestazioni e riavvia l'istanza.

Fase 6: Gestire l'agente

Per avviare, arrestare e controllare lo stato dell'agente, consultaGestisci l'agente.

Gestisci l'agente

L' AWS Ground Station agente offre le seguenti funzionalità per la configurazione, l'avvio, l'arresto, l'aggiornamento, il downgrade e la disinstallazione dell'agente utilizzando gli strumenti di comando Linux integrati.

Argomenti

- AWS Ground Station Configurazione dell'agente
- AWS Ground Station Inizio dell'agente
- AWS Ground Station Agente, fermati
- AWS Ground Station Aggiornamento dell'agente
- AWS Ground Station Downgrade dell'agente
- AWS Ground Station Disinstallazione dell'agente
- AWS Ground Station Stato dell'agente
- AWS Ground Station Informazioni sull'RPM dell'agente

AWS Ground Station Configurazione dell'agente

Vai a`/opt/aws/groundstation/etc`, che dovrebbe contenere un singolo file denominato aws-gs-agent-config .json. Per informazioni, consultare File di configurazione dell'agente.

AWS Ground Station Inizio dell'agente

```
#start
sudo systemctl start aws-groundstation-agent

#check status
systemctl status aws-groundstation-agent
```

Dovrebbe produrre un output che mostri che l'agente è attivo.

```
aws-groundstation-agent.service - aws-groundstation-agent
```

```
Loaded: loaded (/usr/lib/systemd/system/aws-groundstation-agent.service; enabled;
 vendor preset: disabled)
Active: active (running) since Tue 2023-03-14 00:39:08 UTC; 1 day 13h ago
Docs: https://aws.amazon.com/ground-station/
Main PID: 8811 (aws-gs-agent)
CGroup: /system.slice/aws-groundstation-agent.service
##8811 /opt/aws/groundstation/bin/aws-gs-agent production
```

AWS Ground Station Agente, fermati

```
#stop
sudo systemctl stop aws-groundstation-agent

#check status
systemctl status aws-groundstation-agent
```

Dovrebbe produrre un output che mostri che l'agente è inattivo (fermato).

```
aws-groundstation-agent.service - aws-groundstation-agent
Loaded: loaded (/usr/lib/systemd/system/aws-groundstation-agent.service; enabled;
 vendor preset: disabled)
Active: inactive (dead) since Thu 2023-03-09 15:35:08 UTC; 6min ago
Docs: https://aws.amazon.com/ground-station/
Process: 84182 ExecStart=/opt/aws/groundstation/bin/launch-aws-gs-agent (code=exited,
 status=0/SUCCESS)
Main PID: 84182 (code=exited, status=0/SUCCESS)
```

AWS Ground Station Aggiornamento dell'agente

1. Scarica la versione più recente dell'agente. Consultare Scarica l'agente.
2. Arresta l'agente di .

```
#stop
sudo systemctl stop aws-groundstation-agent
```

```
#confirm inactive (stopped) state
systemctl status aws-groundstation-agent
```

3. Aggiorna l'agente.

```
sudo yum update ${MY_RPM_FILE_PATH}

# check the new version has been installed correctly by comparing the agent version
 with the starting agent version
yum info aws-groundstation-agent

# reload the systemd configuration
sudo systemctl daemon-reload

# restart the agent
sudo systemctl restart aws-groundstation-agent

# check agent status
systemctl status aws-groundstation-agent
```

AWS Ground Station Downgrade dell'agente

1. Scarica la versione agente di cui hai bisogno. Consultare Scarica l'agente.
2. Effettua il downgrade dell'agente.

```
# get the starting agent version
yum info aws-groundstation-agent

# stop the agent service
sudo systemctl stop aws-groundstation-agent

# downgrade the rpm
sudo yum downgrade ${MY_RPM_FILE_PATH}

# check the new version has been installed correctly by comparing the agent version
 with the starting agent version
```

```
yum info aws-groundstation-agent

# reload the systemd configuration
sudo systemctl daemon-reload

# restart the agent
sudo systemctl restart aws-groundstation-agent

# check agent status
systemctl status aws-groundstation-agent
```

AWS Ground Station Disinstallazione dell'agente

La disinstallazione dell'agente rinominerà/opt/aws/groundstation/etc/aws-gs-agent-config.json to /opt/aws/groundstation/etc/aws- gs-agent-config .json.rpmsave. La reinstallazione dell'agente sulla stessa istanza scriverà i valori predefiniti per aws-gs-agent-config .json e dovrà essere aggiornato con i valori corretti corrispondenti alle risorse AWS. Consultare File di configurazione dell'agente.

```
sudo yum remove aws-groundstation-agent
```

AWS Ground Station Stato dell'agente

Lo stato dell'agente è attivo (l'agente è in esecuzione) o inattivo (l'agente è fermo).

```
systemctl status aws-groundstation-agent
```

Un esempio di output mostra che l'agente è installato, inattivo (interrotto) e abilitato (avvia il servizio all'avvio).

```
aws-groundstation-agent.service - aws-groundstation-agent
Loaded: loaded (/usr/lib/systemd/system/aws-groundstation-agent.service; enabled;
 vendor preset: disabled)
Active: inactive (dead) since Thu 2023-03-09 15:35:08 UTC; 6min ago
```

```
Docs: https://aws.amazon.com/ground-station/
Process: 84182 ExecStart=/opt/aws/groundstation/bin/launch-aws-gs-agent (code=exited,
 status=0/SUCCESS)
Main PID: 84182 (code=exited, status=0/SUCCESS)
```

AWS Ground Station Informazioni sull'RPM dell'agente

```
yum info aws-groundstation-agent
```

L'output è il seguente:

> **Note**
>
> La «versione» potrebbe essere diversa in base all'ultima versione pubblicata dall'agente.

```
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Installed Packages
Name        : aws-groundstation-agent
Arch        : x86_64
Version     : 1.0.2677.0
Release     : 1
Size        : 51 M
Repo        : installed
Summary     : Client software for AWS Ground Station
URL         : https://aws.amazon.com/ground-station/
License     : Proprietary
Description : This package provides client applications for use with AWS Ground Station
```

Configura l'agente

Dopo aver installato l'agente, è necessario aggiornare il file di configurazione dell'agente all'indirizzo `/opt/aws/groundstation/etc/aws-gs-agent-config.json`.

File di configurazione dell'agente

Esempio

```
{
  "capabilities": [
    "arn:aws:groundstation:eu-central-1:123456789012:dataflow-endpoint-group/
bb6c19ea-1517-47d3-99fa-3760f078f100"
  ],
  "device": {
  "privateIps": [
    "127.0.0.1"
  ],
  "publicIps": [
    "1.2.3.4"
  ],
  "agentCpuCores":
 [ 24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,72,73,74,75,76,77,78,79,80,81
}
```

Suddivisione del campo

capacità

Le funzionalità sono specificate come Dataflow Endpoint Group Amazon Resource Names.

Obbligatorio: True

Formato: String Array

- Valori: capability ARNs → String

Esempi:

```
"capabilities": [
    "arn:aws:groundstation:${AWS::Region}:${AWS::AccountId}:dataflow-endpoint-group/
${DataflowEndpointGroupId}"
]
```

dispositivo

Questo campo contiene campi aggiuntivi necessari per enumerare il EC2 «dispositivo» corrente.

Obbligatorio: True

Formato: oggetto

Membri:

- IP privati
- IP pubblici
- agentCpuCores
- Adattatori di rete

IP privati

Questo campo non è attualmente utilizzato, ma è incluso per casi d'uso futuri. Se non è incluso alcun valore, il valore predefinito è [«127.0.0.1»]

Obbligatorio: falso

Formato: String Array

- Valori: Indirizzi IP → Stringa

Esempio:

```
"privateIps": [
    "127.0.0.1"
],
```

IP pubblici

IP elastico (EIP) per gruppo di endpoint di flussi di dati.

Obbligatorio: True

Formato: String Array

- Valori: Indirizzi IP → Stringa

Esempio:

```
"publicIps": [
    "9.8.7.6"
],
```

agente CPUCores

Questo specifica quali core virtuali sono riservati al aws-gs-agent processo. Vedi i requisiti Pianificazione del core della CPU per impostare questo valore in modo appropriato.

Obbligatorio: True

Formato: Int Array

- Valori: Core Numbers → int

Esempio:

```
"agentCpuCores": [

 24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,72,73,74,75,76,77,78,79,80,81,8
]
```

Adattatori di rete

Corrisponde agli adattatori Ethernet, o alle interfacce collegate, che riceveranno i dati ENIs.

Obbligatorio: falso

Formato: String Array

- Valori: nomi degli adattatori Ethernet (è possibile trovarli eseguendo`ifconfig`)

Esempio:

```
"networkAdapters": [
    "eth0"
]
```

Ottimizza la tua EC2 istanza per migliorarne le prestazioni

> **Note**
>
> Se hai effettuato il provisioning delle risorse AWS utilizzando CloudFormation modelli, questi tuning vengono applicati automaticamente. Se hai utilizzato un'AMI o hai creato manualmente l' EC2 istanza, questi aggiustamenti delle prestazioni devono essere applicati per ottenere le prestazioni più affidabili.

Ricordati di riavviare l'istanza dopo aver applicato qualsiasi ottimizzazione.

Argomenti

- Ottimizza le interruzioni hardware e le code di ricezione: influisce sulla CPU e sulla rete
- Tune Rx interrupt coalescenza: influisce sulla rete
- Tune Rx ring buffer: influisce sulla rete
- Tune CPU C-State: influisce sulla CPU
- Riserva le porte di ingresso: influisce sulla rete
- Riavvio

Ottimizza le interruzioni hardware e le code di ricezione: influisce sulla CPU e sulla rete

Questa sezione configura l'utilizzo del core della CPU di systemd, SMP IRQs, Receive Packet Steering (RPS) e Receive Flow Steering (RFS). Vedi Appendice: Parametri consigliati per l'accordatura interrupt/RPS una serie di impostazioni consigliate in base al tipo di istanza che stai utilizzando.

1. Allontana i processi systemd dai core della CPU dell'agente.
2. Indirizza le richieste di interruzione hardware lontano dai core della CPU dell'agente.
3. Configura RPS per evitare che la coda hardware di una singola scheda di interfaccia di rete diventi un collo di bottiglia nel traffico di rete.
4. Configura RFS per aumentare la frequenza di accesso alla cache della CPU e quindi ridurre la latenza di rete.

Lo `set_irq_affinity.sh` script fornito dall'RPM configura tutto quanto sopra per te. Aggiungilo a crontab, in modo che venga applicato ad ogni avvio:

```
echo "@reboot sudo /opt/aws/groundstation/bin/set_irq_affinity.sh
 '${interrupt_core_list}' '${rps_core_mask}' >> /var/log/user-data.log 2>&1" >>/var/
spool/cron/root
```

- Sostituiscilo `interrupt_core_list` con core riservati al kernel e al sistema operativo, in genere il primo e il secondo insieme a coppie di core hyperthreaded. Questo non dovrebbe sovrapporsi ai core selezionati sopra. (Es: '0,1,48,49' per un'istanza Hyper-Thread da 96 CPU).
- `rps_core_mask`è una maschera di bit esadecimale che specifica quali devono elaborare i pacchetti in entrata, con ogni cifra che CPUs rappresenta 4. CPUs Deve inoltre essere separato da virgole ogni 8 caratteri a partire da destra. Si consiglia di consentire tutto CPUs e lasciare che la memorizzazione nella cache gestisca il bilanciamento.
  - Per visualizzare l'elenco dei parametri consigliati per ogni tipo di istanza, fare riferimento a. Appendice: Parametri consigliati per l'accordatura interrupt/RPS
- Esempio di istanza da 96 CPU:

```
echo "@reboot sudo /opt/aws/groundstation/bin/set_irq_affinity.sh '0,1,48,49'
 'ffffffff,ffffffff,ffffffff' >> /var/log/user-data.log 2>&1" >>/var/spool/cron/root
```

Tune Rx interrupt coalescenza: influisce sulla rete

La coalescenza delle interruzioni aiuta a prevenire l'inondazione del sistema host con troppe interruzioni e aiuta ad aumentare la velocità di trasmissione della rete. Con questa configurazione, i pacchetti vengono raccolti e viene generata una singola interruzione ogni 128 microsecondi. Aggiungi a crontab, in modo che venga applicato ad ogni avvio:

```
echo "@reboot sudo ethtool -C ${interface} rx-usecs 128 tx-usecs 128 >>/var/log/user-
data.log 2>&1" >>/var/spool/cron/root
```

- Sostituisci `interface` con l'interfaccia di rete (adattatore ethernet) configurata per ricevere dati. In genere, `eth0` si tratta dell'interfaccia di rete predefinita assegnata a un' EC2 istanza.

Tune Rx ring buffer: influisce sulla rete

Aumenta il numero di ingressi ad anello per il ring buffer Rx per evitare cadute o sovraccarichi di pacchetti durante le connessioni interrotte. Aggiungi al crontab, in modo che sia impostato correttamente su ogni avvio:

```
echo "@reboot sudo ethtool -G ${interface} rx 16384 >>/var/log/user-data.log 2>&1" >>/
var/spool/cron/root
```

- Sostituisci `interface` con l'interfaccia di rete (adattatore ethernet) configurata per ricevere dati. In genere, `eth0` si tratta dell'interfaccia di rete predefinita assegnata a un' EC2 istanza.
- Se si configura un'istanza `c6i` familiare, è necessario modificare il comando per impostare il ring buffer su`8192`, anziché`16384`.

Tune CPU C-State: influisce sulla CPU

Imposta lo stato C della CPU per evitare l'inattività, che può causare la perdita di pacchetti durante l'avvio di un contatto. Richiede il riavvio dell'istanza.

```
echo "GRUB_CMDLINE_LINUX_DEFAULT=\"console=tty0 console=ttyS0,115200n8
 net.ifnames=0 biosdevname=0 nvme_core.io_timeout=4294967295 intel_idle.max_cstate=1
 processor.max_cstate=1 max_cstate=1\"" >/etc/default/grub
echo "GRUB_TIMEOUT=0" >>/etc/default/grub
grub2-mkconfig -o /boot/grub2/grub.cfg
```

Riserva le porte di ingresso: influisce sulla rete

Riservate tutte le porte nell'intervallo `AwsGroundStationAgentEndpoint` di porte degli indirizzi di ingresso per evitare conflitti con l'utilizzo del kernel. Il conflitto di utilizzo delle porte porterà a problemi di contatto e di consegna dei dati.

```
echo "net.ipv4.ip_local_reserved_ports=${port_range_min}-${port_range_max}" >> /etc/
sysctl.conf
```

- Esempio: `echo "net.ipv4.ip_local_reserved_ports=42000-43500" >> /etc/sysctl.conf`.

Riavvio

Dopo che tutte le regolazioni sono state applicate correttamente, riavvia l'istanza per rendere effettive le ottimizzazioni.

```
sudo reboot
```

Appendice: Parametri consigliati per l'accordatura interrupt/RPS

Questa sezione determina i valori dei parametri consigliati da utilizzare nella sezione di ottimizzazione Tune Hardware Interrupts and Receive Queues - Impact CPU and Network.

Family	Tipo di istanza	$ {interrupt_core_list}	$ {rps_core_mask}
c6i	• c6i.32xlarge	• 0,1,64,65	• ffffffff, ffffffff, ffffffff, ffffffff
c5	• c5.24xlarge • c5.18xlarge • c5.12xlarge	• 0,1,48,49 • 0,1,36,37 • 0,1,24,25	• ffffffff, ffffffff, ffffffff • ff, ffffffff, ffffffff • ffff, ffffffff

Family	Tipo di istanza	$ {interrupt_core_list}	$ {rps_core_mask}
c5n	• c5n.metal • c5n.18xlarge	• 0,1,36,37 • 0,1,36,37	• ff, ffffffff, ffffffff • ff, ffffffff, ffffffff
m5	• m5.24xlarge • m5.12xlarge	• 0,1,48,49 • 0,1,24,25	• ffffffff, ffffffff, ffffffff • ffff, ffffffff
r5	• r5.metal • r5.24xlarge	• 0,1,48,49 • 0,1,48,49	• ffffffff, ffffffff, ffffffff • ffffffff, ffffffff, ffffffff
r5n	• r5n.metal • r5n.24xlarge	• 0,1,48,49 • 0,1,48,49	• ffffffff, ffffffff, ffffffff • ffffffff, ffffffff, ffffffff
g4dn	• g4dn.metal • g4dn.16xlarge • g4dn.12xlarge	• 0,1,48,49 • 0,1,32,33 • 0,1,24,25	• ffffffff, ffffffff, ffffffff • ffffffff, ffffffff • ffff, ffffffff

Family	Tipo di istanza	$ {interrupt_core_list}	$ {rps_core_mask}
p4d	• p4d.24xlarge	• 0,1,48,49	• ffffffff, ffffffff, ffffffff
p3dn	• p3dn.24xlarge	• 0,1,48,49	• ffffffff, ffffffff, ffffffff

Preparati a prendere un contatto DigiF

1. Esamina CPU Core Planning per i flussi di dati desiderati e fornisci un elenco di core che l'agente può utilizzare. Consultare Pianificazione del core della CPU.
2. Esamina il file di configurazione dell' AWS Ground Station agente. Consultare AWS Ground Station Configurazione dell'agente.
3. Verificare che sia stata applicata la necessaria ottimizzazione delle prestazioni. Consultare Ottimizza la tua EC2 istanza per migliorarne le prestazioni.
4. Conferma di seguire tutte le best practice indicate. Consultare Best practice.
5. Conferma che l' AWS Ground Station agente sia stato avviato prima dell'orario di inizio del contatto pianificato tramite:

   ```
   systemctl status aws-groundstation-agent
   ```

6. Verifica che l' AWS Ground Station agente sia in buone condizioni prima dell'orario di inizio del contatto pianificato tramite:

   ```
   aws groundstation get-dataflow-endpoint-group --dataflow-endpoint-group-id
    ${DATAFLOW-ENDPOINT-GROUP-ID} --region ${REGION}
   ```

   Verifica che il `agentStatus` tuo `awsGroundStationAgentEndpoint` sia ATTIVO e che `auditResults` sia SANO.

Best practice

EC2 Le migliori pratiche di Amazon

Segui le EC2 migliori pratiche correnti e assicurati una disponibilità sufficiente di storage dei dati.

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-best-practices.html

Pianificatore Linux

Lo scheduler Linux può riordinare i pacchetti su socket UDP se i processi corrispondenti non sono collegati a un core specifico. Qualsiasi thread che invia o riceve dati UDP deve collegarsi a un core specifico per tutta la durata della trasmissione dei dati.

AWS Ground Station elenco di prefissi gestiti

Si consiglia di utilizzare l'elenco di prefissi `com.amazonaws.global.groundstation` gestito da AWS quando si specificano le regole di rete per consentire la comunicazione dall'antenna. Consulta Working with AWS Managed Prefix Lists per ulteriori informazioni su AWS Managed Prefix Lists.

Limitazione del contatto singolo

AWS Ground Station Agent supporta più flussi per contatto, ma supporta solo un contatto alla volta. Per evitare problemi di pianificazione, non condividere un'istanza tra più gruppi di endpoint di flussi di dati. Se una configurazione a singolo agente è associata a più DFEG diversi ARNs, non riuscirà a registrarsi.

Esecuzione di servizi e processi insieme all'agente AWS Ground Station

Quando si avviano servizi e processi sulla stessa EC2 istanza dell' AWS Ground Station agente, è importante associarli a v CPUs non utilizzati dall' AWS Ground Station agente e dal kernel Linux poiché ciò può causare colli di bottiglia e persino la perdita di dati durante i contatti. Questo concetto di associazione a una v specifica è noto come affinità. CPUs

Core da evitare:

- `agentCpuCores`da File di configurazione dell'agente
- `interrupt_core_list` da Ottimizza le interruzioni hardware e le code di ricezione: influisce sulla CPU e sulla rete.
  - I valori predefiniti possono essere trovati da Appendice: Parametri consigliati per l'accordatura interrupt/RPS

Ad esempio, utilizzando un'**`c5.24xlarge`**istanza

Se hai specificato

```
"agentCpuCores": [24,25,26,27,72,73,74,75]"
```

e sono scappato

```
echo "@reboot sudo /opt/aws/groundstation/bin/set_irq_affinity.sh
'0,1,48,49' 'ffffffff,ffffffff,ffffffff' >> /var/log/user-data.log 2>&1"
>>/var/spool/cron/root
```

quindi evita i seguenti core:

```
0,1,24,25,26,27,48,49,72,73,74,75
```

Servizi di affinitizzazione (systemd)

I servizi appena lanciati verranno automaticamente affinitizzati a quelli menzionati in precedenza. `interrupt_core_list` Se il caso d'uso dei servizi lanciati richiede core aggiuntivi o richiede core meno congestionati, segui questa sezione.

Controlla a quale affinità è attualmente configurato il tuo servizio con il comando:

```
systemctl show --property CPUAffinity <service name>
```

Se vedi un valore vuoto come`CPUAffinity=`, significa che probabilmente utilizzerà i core predefiniti del comando precedente `...bin/set_irq_affinity.sh <using the cores here> ...`

Per sovrascrivere e impostare un'affinità specifica, trova la posizione del file di servizio eseguendo:

```
systemctl show -p FragmentPath <service name>
```

Apri e modifica il file (usando `vinano`, ecc.) e inseriscilo `CPUAffinity=<core list>` nella `[Service]` sezione come segue:

```
[Unit]
...

[Service]
...
CPUAffinity=2,3

[Install]
...
```

Salva il file e riavvia il servizio per applicare l'affinità con:

```
systemctl daemon-reload
systemctl restart <service name>

# Additionally confirm by re-running
systemctl show --property CPUAffinity <service name>
```

Per maggiori informazioni visita: Red Hat Enterprise Linux 8 - Gestione, monitoraggio e aggiornamento del kernel - Capitolo 27. Configurazione delle politiche CPU Affinity e NUMA utilizzando systemd.

Affinitizzazione dei processi (script)

Si consiglia vivamente di affinitizzare manualmente gli script e i processi appena lanciati, poiché il comportamento predefinito di Linux consentirà loro di utilizzare qualsiasi core della macchina.

Per evitare conflitti di base per qualsiasi processo in esecuzione (come python, script bash, ecc.), avvia il processo con:

```
taskset -c <core list> <command>
# Example: taskset -c 8 ./bashScript.sh
```

Se il processo è già in esecuzione, usa comandi come `pidoftop`, o `ps` per trovare l'ID di processo (PID) del processo specifico. Con il PID puoi vedere l'attuale affinità con:

```
taskset -p <pid>
```

e puoi modificarlo con:

```
taskset -p <core mask> <pid>
# Example: taskset -p c 32392 (which sets it to cores 0xc -> 0b1100 -> cores 2,3)
```

Per ulteriori informazioni su taskset, vedere taskset - Linux man page

Risoluzione dei problemi

L'agente non riesce ad avviarsi

L' AWS Ground Station agente potrebbe non avviarsi per diversi motivi, ma lo scenario più comune potrebbe essere un file di configurazione dell'agente configurato in modo errato. Dopo aver avviato l'agente (vediAWS Ground Station Inizio dell'agente) potresti ottenere uno stato come:

```
#agent is automatically retrying a restart
aws-groundstation-agent.service - aws-groundstation-agent
Loaded: loaded (/usr/lib/systemd/system/aws-groundstation-agent.service; enabled;
 vendor preset: disabled)
Active: activating (auto-restart) (Result: exit-code) since Fri 2023-03-10 01:48:14
 UTC; 23s ago
Docs: https://aws.amazon.com/ground-station/
Process: 43038 ExecStart=/opt/aws/groundstation/bin/launch-aws-gs-agent (code=exited,
 status=101)
Main PID: 43038 (code=exited, status=101)

#agent has failed to start
aws-groundstation-agent.service - aws-groundstation-agent
Loaded: loaded (/usr/lib/systemd/system/aws-groundstation-agent.service; enabled;
 vendor preset: disabled)
Active: failed (Result: start-limit) since Fri 2023-03-10 01:50:15 UTC; 13s ago
Docs: https://aws.amazon.com/ground-station/
Process: 43095 ExecStart=/opt/aws/groundstation/bin/launch-aws-gs-agent (code=exited,
 status=101)
Main PID: 43095 (code=exited, status=101)
```

Risoluzione dei problemi

```
sudo journalctl -u aws-groundstation-agent | grep -i -B 3 -A 3 'Loading Config' | tail
 -6
```

potrebbe risultare in un output di:

```
launch-aws-gs-agent[43095]: Running with options Production(ProductionOptions
 { endpoint: None, region: None })
launch-aws-gs-agent[43095]: Loading Config
launch-aws-gs-agent[43095]: System has 96 logical cores
systemd[1]: aws-groundstation-agent.service: main process exited, code=exited,
 status=101/n/a
systemd[1]: Unit aws-groundstation-agent.service entered failed state.
```

Il mancato avvio dell'agente dopo «Loading Config» indica un problema con la configurazione dell'agente. Vedi File di configurazione dell'agente per verificare la configurazione dell'agente.

AWS Ground Station Registri degli agenti

AWS Ground Station L'agente scrive informazioni sull'esecuzione dei contatti, sugli errori e sullo stato di integrità nei file di registro sull'istanza su cui è in esecuzione l'agente. È possibile visualizzare i file di registro connettendosi manualmente a un'istanza.

È possibile visualizzare i log degli agenti nella seguente posizione.

```
/var/log/aws/groundstation
```

Nessun contatto disponibile

La pianificazione dei contatti richiede un AWS Ground Station agente in buona salute. Conferma che il tuo AWS Ground Station agente sia stato avviato e che sia integro interrogando l' AWS Ground Station API tramite: get-dataflow-endpoint-group

```
aws groundstation get-dataflow-endpoint-group --dataflow-endpoint-group-id ${DATAFLOW-
ENDPOINT-GROUP-ID} --region ${REGION}
```

Verifica che il `agentStatus` tuo `awsGroundStationAgentEndpoint` sia ATTIVO e che `auditResults` sia SANO.

Ottenere supporto

Contatta il team di Ground Station tramite AWS Support.

1. Fornisci tutti `contact_id` i contatti interessati. Il AWS Ground Station team non può indagare su un contatto specifico senza queste informazioni.
2. Fornisci dettagli su tutte le procedure di risoluzione dei problemi già intraprese.
3. Fornisci eventuali messaggi di errore rilevati durante l'esecuzione dei comandi nella nostra guida alla risoluzione dei problemi.

Note sulla versione dell'agente

Versione più recente dell'agente

Versione 1.0.3555.0

Data di rilascio: 27/03/2024

Data di fine del supporto: 30/06/2025

Checksum RPM:

- SHA256: `108f3aceb00e5af549839cd766c56149397e448a6e1e1429c89a9eebb6bc0fc1`
- MD5: `65b72fa507fb0af32651adbb18d2e30f`

Modifiche:

- Aggiungi la metrica dell'agente per la versione eseguibile selezionata durante l'avvio del task.
- Aggiungi il supporto per i file di configurazione per evitare versioni eseguibili specifiche quando sono disponibili altre versioni.
- Aggiungi la diagnostica di rete e di routing.
- Funzionalità di sicurezza aggiuntive.
- Risolto il problema per cui alcuni errori di segnalazione delle metriche venivano scritti nel file di registro stdout/journal anziché nel file di registro.
- Gestisci con garbo gli errori di socket irraggiungibili dalla rete.
- Misura la perdita di pacchetti e la latenza tra gli agenti di origine e di destinazione.
- Rilascia la aws-gs-datapipe versione 2.0 per supportare nuove funzionalità del protocollo e la possibilità di aggiornare in modo trasparente i contatti al nuovo protocollo.

Versioni obsolete degli agenti

Versione 1.0.2942.0

Data di rilascio: 26/06/2023

Data di fine del supporto: 31/05/2024

Checksum RPM:

- SHA256: `7d94b642577504308a58bab28f938507f2591d4e1b2c7ea170b77bea97b5a9b6`
- MD5: `661ff2b8f11aba5d657a6586b56e0d8f`

Modifiche:

- Sono stati aggiunti i log degli errori relativi all'aggiornamento di Agent RPM su disco e per rendere effettive le modifiche è necessario il riavvio dell'agente.
- È stata aggiunta la convalida dell'ottimizzazione della rete per garantire che le fasi di ottimizzazione della guida per l'utente di Agent siano seguite e applicate correttamente.
- Risolto il bug che causava avvisi errati nei log di Agent sull'archiviazione dei log.
- Rilevamento migliorato della perdita di pacchetti.
- Installazione aggiornata dell'agente per impedire l'installazione o l'aggiornamento dell'RPM se l'agente è già in esecuzione.

Versione 1.0.2716.0

Data di rilascio: 15/03/2023

Data di fine del supporto: 31/05/2024

Checksum RPM:

- SHA256: `cb05b6a77dfcd5c66d81c0072ac550affbcefefc372cc5562ee52fb220844929`
- MD5: `65266490c4013b433ec39ee50008116c`

Modifiche:

- Abilita il caricamento dei log quando l'agente riscontra errori durante l'esecuzione delle attività.
- Risolve il bug di compatibilità con Linux negli script di ottimizzazione della rete forniti.

Versione 1.0.2677.0

Data di rilascio: 15/02/2023

Data di fine del supporto: 31/05/2024

Checksum RPM:

- SHA256: `77cfe94acb00af7ca637264b17c9b21bd7afdc85b99dffdd627aec9e99397489`
- MD5: `b8533be7644bb4d12ab84de21341adac`

Modifiche:

- Prima versione di Agent disponibile a livello generale.

Convalida dell'installazione RPM

La versione RPM più recente, l' MD5 hash convalidato da RPM e SHA256 l'hash che utilizza sha256sum sono mostrati di seguito. Questi valori, combinati, possono essere utilizzati per convalidare la versione RPM utilizzata per l'agente della stazione di terra.

Versione più recente dell'agente

Versione 1.0.3555.0

Data di rilascio: 27/03/2024

Data di fine del supporto: 30/06/2025

Checksum RPM:

- SHA256: `108f3aceb00e5af549839cd766c56149397e448a6e1e1429c89a9eebb6bc0fc1`
- MD5: `65b72fa507fb0af32651adbb18d2e30f`

Modifiche:

- Aggiungi la metrica dell'agente per la versione eseguibile selezionata durante l'avvio del task.
- Aggiungi il supporto per i file di configurazione per evitare versioni eseguibili specifiche quando sono disponibili altre versioni.
- Aggiungi la diagnostica di rete e di routing.
- Funzionalità di sicurezza aggiuntive.
- Risolto il problema per cui alcuni errori di segnalazione delle metriche venivano scritti nel file di registro stdout/journal anziché nel file di registro.
- Gestisci con garbo gli errori di socket irraggiungibili dalla rete.
- Misura la perdita di pacchetti e la latenza tra gli agenti di origine e di destinazione.
- Rilascia la aws-gs-datapipe versione 2.0 per supportare nuove funzionalità del protocollo e la possibilità di aggiornare in modo trasparente i contatti al nuovo protocollo.

Verifica l'RPM

Gli strumenti necessari per verificare questa installazione RPM sono:

- sha256 sum
- giri/min

Entrambi gli strumenti sono disponibili di default su Amazon Linux 2. Questi strumenti ti aiuteranno a verificare che l'RPM che stai utilizzando sia la versione corretta. Per prima cosa scaricate l'RPM più recente dal bucket S3 (Scarica l'agenteper istruzioni su come scaricare l'RPM, consultate la sezione RPM). Una volta scaricato questo file, ci saranno alcune cose da controllare:

- Calcola la somma sha256sum del file RPM. Esegui la seguente azione dalla riga di comando dell'istanza di calcolo che stai utilizzando:

  ```
  sha256sum aws-groundstation-agent.rpm
  ```

  Prendi questo valore e confrontalo con la tabella precedente. Ciò dimostra che il file RPM scaricato è un file valido da utilizzare che AWS Ground Station ha distribuito ai clienti. Se gli hash non corrispondono, non installare l'RPM ed eliminalo dall'istanza di calcolo.
- Controlla anche l' MD5 hash del file, per assicurarti che l'RPM non sia stato compromesso. Per fare ciò, utilizzate lo strumento da riga di comando RPM eseguendo il seguente comando:

  ```
  rpm -Kv ./aws-groundstation-agent.rpm
  ```

  Verifica che l' MD5 hash qui elencato sia lo stesso dell' MD5 hash della versione riportata nella tabella precedente. Una volta che entrambi questi hash sono stati convalidati rispetto a questa tabella elencata in AWS Docs, il cliente può essere certo che l'RPM scaricato e installato sia la versione sicura e senza compromessi dell'RPM.

Cronologia dei documenti per l' AWS Ground Station Agent User Guide

La tabella seguente descrive le modifiche importanti in ogni versione della AWS Ground Station Agent User Guide.

Modifica	Descrizione	Data
Aggiornamento della documentazione	Supporto rimosso per la famiglia di istanze precedente: m4.	30 settembre 2024
Aggiornamento della documentazione	È stato aggiunto un commento sul mantenimento della sottorete e dell' EC2 istanza Amazon nella stessa zona di disponibilità in Agent Requirements.	18 luglio 2024
Aggiornamento della documentazione	Suddividi l' AWS Ground Station agente in una propria guida per l'utente. Per le modifiche precedenti, consulta: Cronologia dei documenti per la guida per l'utente di AWS Ground Station.	18 luglio 2024

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.