

Entwicklerhandbuch

GameLift Amazon-Streams



GameLift Amazon-Streams: Entwicklerhandbuch

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Die Handelsmarken und Handelsaufmachung von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, durch die Kunden irregeführt werden könnten oder Amazon in schlechtem Licht dargestellt oder diskreditiert werden könnte. Alle anderen Handelsmarken, die nicht Eigentum von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise zu Amazon gehören oder nicht, mit Amazon verbunden sind oder von Amazon gesponsert werden.

Table of Contents

Was ist Amazon GameLift Streams?	1
Feature	1
Erste Schritte mit Amazon GameLift Streams	2
Zugriff auf Amazon GameLift Streams	2
Nutzungsbedingungen	3
Einrichtung	4
Melden Sie sich an für ein AWS-Konto	4
Erstellen eines Benutzers mit Administratorzugriff	5
Holen Sie sich programmatischen Zugriff	6
Laden Sie das Web-SDK herunter	7
Laden Sie das herunter AWS CLI	7
Richten Sie Abrechnungsbenachrichtigungen ein	8
Erste Schritte	9
Eine Konfiguration auswählen	9
Ausgangspunkt	10
Kostenoptimierungen	11
Entscheiden Sie sich für eine Konfiguration	12
Wie sich Ihre Konfigurationsentscheidungen auf die nächsten Schritte auswirken	14
Nächste Schritte	14
Konfigurationsoptionen	15
Laufzeitumgebungen	15
Streamen Sie Klassen	16
Kompatibilität mit Laufzeitumgebung und Stream-Klasse	21
Dein erster Stream	22
Voraussetzungen	23
Schritt 1: Laden Sie Ihre Anwendung in einen Amazon S3 S3-Bucket hoch	24
Schritt 2: Konfigurieren Sie Ihre Anwendung für Amazon GameLift Streams	26
Schritt 3: Verwalten Sie, wie Amazon GameLift Streams Ihre Anwendung streamt	30
Schritt 4: Testen Sie Ihren Stream in Amazon GameLift Streams	37
Schritt 5: Aufräumen (nicht überspringen)	39
Verwalte deine Streams	41
Die wichtigsten Konzepte	41
Anwendungen	43
Bevor Sie hochladen	43

Laden Sie Ihre Anwendung in einen Amazon S3 S3-Bucket hoch	44
Erstellen einer Anwendung	45
Bearbeiten Sie eine Anwendung	52
Löschen einer Anwendung	55
Berechtigungsrichtlinie für das Anwendungsprotokoll-Bucket	57
Verknüpfte Stream-Gruppen	58
Stream-Gruppen	59
Über die Stream-Kapazität	60
Kapazität und Servicekontingenten	62
Über Standorte	64
Erstellen Sie eine Stream-Gruppe	64
Bearbeiten Sie die allgemeinen Einstellungen	77
Kapazität bearbeiten	79
Verhalten beim Herunterskalieren der Kapazität	81
Fügen Sie Standorte zu einer Stream-Gruppe hinzu	81
Standorte in einer Stream-Gruppe entfernen	83
Löschen Sie eine Stream-Gruppe	84
Verknüpfte Anwendungen	86
Lebenszyklus einer Stream-Gruppe	86
Wartung der Stream-Gruppe	87
Stream-Gruppen für mehrere Anwendungen	87
Einschränkungen und Anforderungen	88
Über Standardanwendungen	88
Standardanwendung ändern	89
Eine Anwendung verknüpfen	91
Verknüpfung mit einer Anwendung aufheben	93
Sessions streamen	95
Über Stream-Sitzungen	95
Einen Stream in der Konsole testen	96
Lebenszyklus einer Stream-Sitzung	97
Timeout-Werte, die sich auf Stream-Sitzungen auswirken	100
Beenden einer Stream-Sitzung	101
erneute Verbindung zu einer Stream-Sitzung herstellen	101
Exportieren Sie Stream-Sitzungsdateien	102
Funktionsweise	102
Auswirkung auf die Kosten	103

Dateien exportieren (Konsole)	103
Dateien exportieren (CLI)	104
Backend-Service und WebClient von Amazon GameLift Streams	107
Unterstützte Browser und Eingaben	107
Bekannte Probleme	109
Einschränkungen	109
IPv6 Unterstützung	110
Erforderliche Anschlüsse	110
Einrichtung eines Webservers und Clients	111
Voraussetzungen	111
Laden Sie das Web-SDK herunter	111
Richten Sie Ihre Streaming-Ressourcen ein	112
Richten Sie einen Backend-Server ein	112
Starten Sie einen WebClient	113
Bereinigen Sie die Streaming-Ressourcen	114
Passen Sie das Erscheinungsbild des Streams an	115
Ladebildschirm	115
Präferenz für das Gebietsschema	116
Umgang mit Mausbewegungen	117
Eingabemodi für die Maus	117
Zeigersperre	118
Bewährte Methoden	119
Datenkanal-Kommunikation	120
Features	120
Datenkanäle verwenden	120
Auf der Client-Seite	121
Auf der Anwendungsseite	121
Checkliste für den Start	125
Das Amazon GameLift Streams-Team benachrichtigen	125
Kompatibilitäts- und Leistungstests	125
Reservierung von Kapazitäten	125
Leistungstests im großen Maßstab	126
Einrichtung vor dem Start	126
Zusätzliche Tipps	126
Benötigen Sie weitere Unterstützung?	127
Sicherheit	128

Datenschutz	129
Verschlüsselung im Ruhezustand	131
Verschlüsselung während der Übertragung	131
Schutz von Endbenutzer-Streams	131
Sitzungsisolierung in Linux-Stream-Klassen	132
Sitzungsisolierung in Windows-Stream-Klassen	132
Verwaltung von Verschlüsselungsschlüsseln	133
Datenschutz für den Datenverkehr zwischen Netzwerken	133
Identitäts- und Zugriffsverwaltung	133
Zielgruppe	134
Authentifizierung mit Identitäten	134
Verwalten des Zugriffs mit Richtlinien	136
So funktioniert Amazon GameLift Streams mit IAM	137
Beispiele für identitätsbasierte Richtlinien	143
Fehlerbehebung	146
Compliance-Validierung	147
Ausfallsicherheit	148
Infrastruktursicherheit	148
Wiederverwendung und Mehrmandantenfähigkeit	149
Schnittstellen-VPC-Endpunkte	150
Konfigurations- und Schwachstellenanalyse	152
Bewährte Methoden für die Gewährleistung der Sicherheit	153
Überwachung von Amazon GameLift Streams	154
Überwachen Sie mit CloudWatch	154
Kapazität und Nutzung der Stream-Gruppe	155
Leistung und Ressourcennutzung der Stream-Gruppe	156
Status des Streams	157
Kundenbindung	158
Datenkanäle	158
Protokollierung von CloudTrail-API-Aufrufen mit	159
Amazon GameLift Streams-Datenereignisse in CloudTrail	161
Amazon GameLift Streams-Verwaltungereignisse in CloudTrail	162
Beispiele für Amazon GameLift Streams-Ereignisse	163
Leistungsstatistiken in Echtzeit	167
Erhalten Sie Leistungsstatistiken	168
Referenz zu Leistungsstatistiken	170

Fehlerbehebung	172
Zugriff verweigert	172
Probleme mit der Anwendung	173
Vorläufige Prüfungen	173
Probleme mit Proton	174
Anwendungsprobleme aufgrund der Bildschirmauflösung	174
Die Anwendung wird beim Start der Stream-Sitzung beendet	174
Die Unreal Engine-Anwendung stürzt ab oder erfordert zusätzliche Abhängigkeiten	175
Leistungsprobleme	176
Die Spielleistung wird reduziert, wenn auf Amazon GameLift Streams gestreamt wird	176
Bei Windows-Anwendungen treten langsame Ladezeiten oder ruckelige Probleme auf	176
Probleme mit der Stream-Konnektivität	182
Probleme mit der Stream-Eingabe	183
Allgemeine Problembehebung bei Eingaben	183
Gamepad- und Mikrofoneingänge funktionieren in nativen Linux-Anwendungen nicht	184
Die Tasteneingabe scheint auf dem macOS-Client hängen zu bleiben	184
Mausbewegungen verhalten sich in Amazon GameLift Streams anders	185
Probleme mit der Stream-Sitzung	185
Die Stream-Sitzung wird nicht gestartet	185
Die Stream-Sitzung wurde beendet	186
Kompatibilität mit Proton	187
Allgemeine Schritte zum Testen und zur Fehlerbehebung	187
Bekannte Probleme mit Proton	187
Richten Sie einen lokalen Computer ein	188
Richten Sie einen Remote-Computer ein	190
Fehlerbehebung bei Proton	195
Profilierung der Leistung von Unreal Engine	199
Regionen, Kontingente und Beschränkungen	202
AWS-Regionen und abgelegene Standorte	202
Service-Endpunkte	203
Entfernte Standorte	203
Unterstützte Standorte nach Stream-Klasse	205
Servicekontingente	206
Servicekontingente	206
API-Ratenlimits	216
Andere Einschränkungen	217

Nutzung und Rechnungen	219
Überprüfen Sie Ihre Amazon GameLift Streams-Rechnungen und Ihre Nutzung	219
Bewährte Methoden zur Verwaltung der Amazon GameLift Streams-Kosten	220
Erstellen Sie Abrechnungsbenachrichtigungen, um die Nutzung zu überwachen	220
Skalieren Sie Stream-Gruppen auf eine Kapazität von Null	220
Löschen Sie die ursprünglichen Anwendungsdateien	221
	ccxxii

Was ist Amazon GameLift Streams?

Mit Amazon GameLift Streams können Spieleverlage und andere Anbieter Streaming-Erlebnisse auf Abruf und mit geringer Latenz für Spieler und Zuschauer auf der ganzen Welt anbieten. Amazon GameLift Streams verwendet seine eigene Streaming-Technologie in Kombination mit der AWS globalen Infrastruktur, um Anwendungsstreaming in großem Umfang zu betreiben und aufrechtzuerhalten. Verlage haben die Flexibilität, sowohl On-Demand-Streaming-Ressourcen als auch reservierte Streaming-Ressourcen bereitzustellen, um Kapazität und Kosten effektiv zu verwalten.

Themen

- [Feature](#)
- [Erste Schritte mit Amazon GameLift Streams](#)
- [Zugriff auf Amazon GameLift Streams](#)

Feature

Amazon GameLift Streams bietet die folgenden Hauptfunktionen:

- Streaming-Technologie, die Echtzeit-Spielerlebnisse mit minimaler player-to-cloud Latenz auf jedem Gerät mit einem Browser bietet, das AWS weltweit präsent ist.
- Nahtloses Spielen von Spielen mit HD-Auflösung (1080p) und 60 Bildern pro Sekunde, ohne dass Downloads erforderlich sind. So wird jedes browserbasierte Gerät zu einem leistungsstarken Spielautomaten.
- Skalierungstools zur Anpassung Ihrer Streaming-Kapazität an die Kundennachfrage. Mit diesen Tools können Sie beispielsweise die Kosten für das Streamen von Spielen im Rahmen halten und gleichzeitig genügend Kapazität aufrechterhalten, um neue Spieler schnell an Stream-Sitzungen teilnehmen zu lassen.
- Stream-Leistungsanalyse mithilfe der Amazon GameLift Streams-Konsole, um Metriken zu verfolgen, Stream-Logs einzusehen und Daten zur Stream-Ressourcennutzung zu überprüfen.
- Direktes Streaming von Windows- und Linux-basierten Spielen mit geringen bis keinen Änderungen.
- Das Amazon GameLift Streams SDK hilft Ihnen bei der Integration Ihrer vorhandenen Identitätsdienste, Storefront- und Client-Anwendungen.

Erste Schritte mit Amazon GameLift Streams

Wenn Sie Amazon GameLift Streams zum ersten Mal nutzen, empfehlen wir Ihnen, mit den folgenden Themen zu beginnen:

- [Amazon GameLift Streams als Entwickler einrichten](#) deckt einmalige Einrichtungsaufgaben ab, einschließlich der Beantragung eines AWS-Konto Benutzerzugriffs und der Einrichtung der Software, die Sie für die Entwicklung mit Amazon GameLift Streams benötigen.
- [Ihren ersten Stream in Amazon GameLift Streams starten](#) führt Sie durch die wichtigsten Schritte im Content-Streaming-Workflow. Ausgehend von Ihren Inhalten, z. B. einem Spiel-Build, stellen Sie Amazon GameLift Streaming-Cloud-Ressourcen bereit und initiieren eine Streaming-Sitzung.

Zugriff auf Amazon GameLift Streams

Sie können Ihre Anwendungsinhalte und Streaming-Ressourcen mit den folgenden Tools erstellen, darauf zugreifen und sie verwalten:

- AWS-Managementkonsole — Bietet eine Weboberfläche, mit der Sie Ihre Amazon GameLift Streams-Anwendungen und Stream-Gruppen erstellen und verwalten können.
- AWS Command Line Interface (AWS CLI) — Stellt Befehle für eine Vielzahl von Befehlen bereit AWS-Services und wird unter Windows, Mac und Linux unterstützt. Weitere Informationen zu diesem Tool finden Sie auf der [AWS Command Line Interface Seite](#).
- AWS SDK — Stellt sprachspezifische Informationen bereit APIs und kümmert sich um Verbindungsdetails wie die Berechnung von Signaturen, die Bearbeitung von Wiederholungsversuchen von Anfragen und die Fehlerbehandlung. Dokumentation zur Amazon GameLift Streams-Service-API finden Sie in der [Amazon GameLift Streams-API-Referenz](#). Weitere allgemeine Informationen zum AWS SDK finden Sie unter [Tools to Build On AWS](#).

Weitere Informationen zur Unterstützung finden AWS-Regionen Sie unter[Regionen, Kontingente und Beschränkungen](#).

Nutzungsbedingungen für Amazon GameLift Streams

Bevor Sie Amazon GameLift Streams verwenden, stellen Sie sicher, dass Sie alle geltenden rechtlichen Anforderungen einhalten können, einschließlich der Lizenzbedingungen, die für die Anwendungen gelten, die Sie streamen möchten, und für die Standorte, an denen Sie streamen möchten.

- Weitere Informationen zu den AWS Anforderungen finden Sie in Abschnitt 43 der [AWS Servicebedingungen](#).
- Weitere Informationen zu den Service Level Agreements finden Sie im [Amazon GameLift Streams Service Level Agreement](#).

Amazon GameLift Streams als Entwickler einrichten

Um mit der Nutzung des Amazon GameLift Streams-Service für Ihre Projekte zu beginnen, führen Sie diese grundlegenden Einrichtungsaufgaben aus. Wenn Sie unter diesem Konto bereits einen Benutzer AWS-Konto und einen Benutzer haben, den Sie mit Amazon GameLift Streams verwenden möchten, können Sie zu wechseln[Laden Sie das Web-SDK herunter](#).

Weitere Informationen darüber, was Sie mit einem machen können AWS-Konto, finden Sie unter [Erste Schritte mit AWS](#).

Nachdem Sie diese Einrichtungsaufgaben abgeschlossen haben, empfehlen wir Ihnen, mit dem Tutorial fortzufahren [Ihren ersten Stream in Amazon GameLift Streams starten](#) und es Schritt für Schritt durchzugehen. Es behandelt den gesamten Arbeitsablauf für das Streamen Ihrer Inhalte in einem Webclient.

Themen

- [Melden Sie sich an für ein AWS-Konto](#)
- [Erstellen eines Benutzers mit Administratorzugriff](#)
- [Holen Sie sich programmatischen Zugriff](#)
- [Laden Sie das Amazon GameLift Streams Web SDK herunter](#)
- [Laden Sie das herunter AWS CLI](#)
- [Richten Sie Abrechnungsbenachrichtigungen ein](#)

Melden Sie sich an für ein AWS-Konto

Wenn Sie noch keine haben AWS-Konto, führen Sie die folgenden Schritte aus, um eine zu erstellen.

Um sich für eine anzumelden AWS-Konto

1. Öffnen Sie <https://portal.aws.amazon.com/billing/die Anmeldung>.
2. Folgen Sie den Online-Anweisungen.

Während der Anmeldung erhalten Sie einen Telefonanruf oder eine Textnachricht und müssen einen Verifizierungscode über die Telefontasten eingeben.

Wenn Sie sich für eine anmelden AWS-Konto, Root-Benutzer des AWS-Kontos wird eine erstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Als bewährte

Sicherheitsmethode weisen Sie einem Administratorbenutzer Administratorzugriff zu und verwenden Sie nur den Root-Benutzer, um [Aufgaben auszuführen, die Root-Benutzerzugriff erfordern.](#)

AWS sendet Ihnen nach Abschluss des Anmeldevorgangs eine Bestätigungs-E-Mail. Du kannst jederzeit deine aktuellen Kontoaktivitäten einsehen und dein Konto verwalten, indem du zu <https://aws.amazon.com/> gehst und Mein Konto auswählst.

Erstellen eines Benutzers mit Administratorzugriff

Nachdem Sie sich für einen angemeldet haben AWS-Konto, sichern Sie Ihren Root-Benutzer des AWS-Kontos AWS IAM Identity Center, aktivieren und erstellen Sie einen Administratorbenutzer, sodass Sie den Root-Benutzer nicht für alltägliche Aufgaben verwenden.

Sichern Sie Ihre Root-Benutzer des AWS-Kontos

1. Melden Sie sich [AWS-Managementkonsole](#) als Kontoinhaber an, indem Sie Root-Benutzer auswählen und Ihre AWS-Konto E-Mail-Adresse eingeben. Geben Sie auf der nächsten Seite Ihr Passwort ein.

Hilfe bei der Anmeldung mit dem Root-Benutzer finden Sie unter [Anmelden als Root-Benutzer](#) im AWS-Anmeldung Benutzerhandbuch zu.

2. Aktivieren Sie die Multi-Faktor-Authentifizierung (MFA) für den Root-Benutzer.

Anweisungen finden Sie unter [Aktivieren eines virtuellen MFA-Geräts für Ihren AWS-Konto Root-Benutzer \(Konsole\)](#) im IAM-Benutzerhandbuch.

Erstellen eines Benutzers mit Administratorzugriff

1. Aktivieren Sie das IAM Identity Center.

Anweisungen finden Sie unter [Aktivieren AWS IAM Identity Center](#) im AWS IAM Identity Center Benutzerhandbuch.

2. Gewähren Sie einem Administratorbenutzer im IAM Identity Center Benutzerzugriff.

Ein Tutorial zur Verwendung von IAM-Identity-Center-Verzeichnis als Identitätsquelle finden Sie IAM-Identity-Center-Verzeichnis im Benutzerhandbuch unter [Benutzerzugriff mit der Standardeinstellung konfigurieren](#). AWS IAM Identity Center

Anmelden als Administratorbenutzer

- Um sich mit Ihrem IAM-Identity-Center-Benutzer anzumelden, verwenden Sie die Anmelde-URL, die an Ihre E-Mail-Adresse gesendet wurde, als Sie den IAM-Identity-Center-Benutzer erstellt haben.

Hilfe bei der Anmeldung mit einem IAM Identity Center-Benutzer finden Sie [im AWS-Anmeldung Benutzerhandbuch unter Anmeldung beim AWS Access-Portal](#).

Weiteren Benutzern Zugriff zuweisen

- Erstellen Sie im IAM-Identity-Center einen Berechtigungssatz, der den bewährten Vorgehensweisen für die Anwendung von geringsten Berechtigungen folgt.

Anweisungen hierzu finden Sie unter [Berechtigungssatz erstellen](#) im AWS IAM Identity Center Benutzerhandbuch.

- Weisen Sie Benutzer einer Gruppe zu und weisen Sie der Gruppe dann Single Sign-On-Zugriff zu.

Eine genaue Anleitung finden Sie unter [Gruppen hinzufügen](#) im AWS IAM Identity Center Benutzerhandbuch.

Holen Sie sich programmatischen Zugriff

Zusätzlich zu Ihren Benutzeranmeldedaten für die benötigen Sie Anmeldeinformationen für den AWS-Managementkonsole programmatischen Zugriff, z. B. wenn Sie mit dem AWS Command Line Interface () arbeiten. AWS CLI Programmatische Anmeldeinformationen bestehen aus einem zweiteiligen Satz von Zugriffsschlüsseln. Verwenden Sie eine der folgenden Methoden, um Ihre Zugriffsschlüssel zu generieren:

- Methode 1 — Wenn Sie einen Administratorbenutzer verwenden, der mit dem IAM Identity Center erstellt wurde, finden Sie unter [Abrufen von IAM-Rollenanmeldedaten für den AWS CLI Zugriff](#) Informationen zur Generierung temporärer Sicherheitsanmeldedaten für den kurzfristigen Zugriff AWS auf Ressourcen. Wenn Sie diese Anweisungen befolgen, stellen Sie sicher, dass Sie über die URL des AWS Zugriffsportals Ihres Kontos mit Ihrem Administratorbenutzernamen und Passwort angemeldet sind (nicht mit Ihrem Root-Benutzer).

- Methode 2 — Wenn Sie einen bestehenden IAM-Benutzer verwenden und noch nicht auf das IAM Identity Center umgestellt haben, finden Sie weitere Informationen unter [Zugriffsschlüssel für IAM-Benutzer \(Konsole\) verwalten, um langfristige Anmeldeinformationen für Ihren Benutzer zu generieren.](#)

Note

Es hat sich bewährt, temporäre Anmeldeinformationen anstelle von langfristigen Zugriffsschlüsseln zu verwenden. Temporäre Anmeldeinformationen bestehen aus einer Zugriffsschlüssel-ID, einem geheimen Zugriffsschlüssel und einem Sicherheits-Token, das angibt, wann die Anmeldeinformationen ablaufen. Weitere Informationen finden Sie unter [Bewährte Methoden für die Verwaltung von AWS Zugriffsschlüsseln](#) in der Allgemeine AWS-Referenz.

Laden Sie das Amazon GameLift Streams Web SDK herunter

Sie können ohne zusätzliche Materialien loslegen, indem Sie das Streaming-Erlebnis in der Konsole nutzen. Wir empfehlen dies als Ausgangspunkt, da Sie so bewerten können, wie Ihre Anwendung in den Amazon GameLift Streams abschneidet, ohne zusätzliche Infrastruktur einrichten zu müssen. Weitere Informationen finden Sie unter [Erste Schritte mit Amazon GameLift Streams](#).

Wenn Sie bereit sind, Ihre eigene Amazon GameLift Streams-Integration zu erstellen, laden Sie das Amazon GameLift Streams Web SDK herunter, das im Abschnitt Ressourcen der [Produktseite Erste Schritte](#) verfügbar ist. Amazon GameLift Streams ist so konzipiert, dass es in Ihre Webanwendungen integriert werden kann. Sie müssen unser JavaScript basiertes Web-SDK integrieren, um Streaming von Ihrer Website oder browserbasierten Anwendungen aus einzurichten. Der Download enthält auch einen Beispiel-Webserver, der den Amazon GameLift Streams-Service verwendet, und einen Beispiel-Webclient für die Verbindung zu Streams.

Weitere Informationen zur Einrichtung Ihrer eigenen Amazon GameLift Streams-Lösung finden Sie unter [Backend-Service und Webclient von Amazon GameLift Streams](#).

Laden Sie das herunter AWS CLI

Um Amazon GameLift Streams mit Ihren Inhalten zu verwenden, empfehlen wir Ihnen, die AWS Command Line Interface (AWS CLI) zu erwerben. Das AWS CLI ist ein Open-Source-Tool, das Ihnen

die gleiche AWS SDK-Funktionalität bietet, indem es Befehle von einem Terminalprogramm aus ausführt.

1. Laden Sie die neueste Version von AWS CLI für Ihr Betriebssystem herunter und installieren Sie sie. Diese [Installationsanweisungen finden Sie](#) im AWS Command Line Interface Benutzerhandbuch.
2. Konfigurieren Sie das Tool mit Ihren Benutzerzugangsdaten und anderen Einstellungen, wie unter [Einrichten](#) von beschrieben AWS CLI. Bei dieser Konfiguration müssen Sie Ihre Anmeldeinformationen und andere Einstellungen nicht bei jedem Befehl explizit angeben.
3. Verwenden Sie den folgenden Befehl, um Ihre Installation zu überprüfen und eine Liste der verfügbaren Amazon GameLift Streams-Befehle abzurufen:

```
aws gameliftstreams help
```

Richten Sie Abrechnungsbenachrichtigungen ein

Für eine Stream-Gruppe fallen Kosten pro aktiver Stream-Kapazität pro Sekunde an. Informationen dazu, wie Sie sicherstellen können, dass Ihre Kosten und Nutzung im Rahmen Ihres Budgets bleiben, finden Sie unter [Erstellen Sie Abrechnungsbenachrichtigungen, um die Nutzung zu überwachen](#).

Erste Schritte mit Amazon GameLift Streams

Dieser Abschnitt kann Ihnen helfen, erfolgreich mit dem Streamen Ihrer Anwendungen und Spiele über Amazon GameLift Streams zu beginnen. Die Themen in diesem Abschnitt behandeln den end-to-end Prozess — vom Hochladen Ihrer Anwendung auf Amazon GameLift Streams bis hin zum Testen der Leistung Ihrer Inhalte in einem Stream. Außerdem werden wichtige Schritte behandelt, die Sie bei der Vorbereitung auf das Streaming unterstützen, wie z. B. die Auswahl der richtigen Laufzeit- und Stream-Class-Konfiguration zur Optimierung von Leistung und Kosten.

Themen

- [Konfiguration in Amazon GameLift Streams auswählen](#)
- [Konfigurationsoptionen](#)
- [Ihren ersten Stream in Amazon GameLift Streams starten](#)

Konfiguration in Amazon GameLift Streams auswählen

Dieser Leitfaden kann Ihnen helfen, die optimale Laufzeitumgebung und die optimalen Konfigurationseinstellungen für das Streamen Ihrer Anwendungen und Spiele über Amazon GameLift Streams auszuwählen. Die Konfigurationseinstellungen wirken sich direkt auf die Leistung Ihrer Inhalte und die Kosten aus, die mit deren Ausführung in Amazon GameLift Streams verbunden sind. Es gibt mehrere Optionen, um eine Vielzahl von Anwendungen und eine hohe Grafikqualität zu unterstützen.

Die vollständige Liste der Konfigurationsoptionen finden Sie unter[Konfigurationsoptionen](#).

Die folgenden Schlüsselbegriffe können Ihnen helfen zu verstehen, wie diese Konfigurationsoptionen zusammenwirken:

- Laufzeiten beziehen sich auf das zugrunde liegende Betriebssystem und die Softwareumgebung, in der Ihre Anwendung auf Amazon GameLift Streams ausgeführt wird. Die wichtigsten Optionen für die Laufzeitumgebung sind Windows, Linux und Proton.
- Stream-Klassen stellen die verschiedenen Ressourcenkonfigurationen dar, die in Amazon GameLift Streams verfügbar sind und sich in Betriebssystem, CPU, GPU, RAM und anderen Spezifikationen unterscheiden. Die Stream-Klasse ist eine Konfigurationsoption einer Stream-Gruppe, die sowohl die einer Stream-Sitzung zugewiesenen Hardwareressourcen als auch das

Tenancy-Modell (wie viele gleichzeitige Streams auf einer einzelnen virtuellen Maschine ausgeführt werden können) definiert.

- Multi-Tenancy ermöglicht es mehreren Benutzern, dieselben zugrunde liegenden Hardwareressourcen gemeinsam zu nutzen. Dies kann eine kostengünstige Option für Anwendungen sein, für die keine maximale Hardwarekapazität erforderlich ist. Eine Stream-Klasse mit Multi-Tenancy kann mehrere Streams zum Preis einer Ressource hosten. Stream-Klassen „High“ haben eine 1:2 -Tenancy, während Stream-Klassen „Ultra“ eine Tenancy haben.

Bei der Einrichtung Ihrer Amazon GameLift Streams-Konfiguration bestimmt die von Ihnen gewählte Laufzeitumgebung die spezifischen Stream-Klassenoptionen, die kompatibel und für Sie verfügbar sind. Die Anpassung der Anforderungen Ihrer Anwendung an die richtige Laufzeitumgebung und Stream-Klasse ist der Schlüssel zur Optimierung der Leistung und Kosteneffizienz in Amazon GameLift Streams.

Die Kosten für das Streamen hängen von der Stream-Klasse ab. Eine detaillierte Kostenliste finden Sie auf der [Seite mit den Amazon GameLift Streams-Preisen](#).

Ausgangspunkt

Abhängig von Ihrer Anwendung sind diese gute Ausgangspunkte, um mit dem Streaming zu beginnen. Später können Sie weitere Konfigurationsoptionen ausprobieren, um die Kosten zu optimieren.

Für Windows-Anwendungen

Wir empfehlen die Verwendung der Microsoft Windows Server 2022 Base-Laufzeitumgebung für Microsoft Windows-Anwendungen. Für diese Laufzeit sind vier Hardwarekonfigurationen verfügbar: die NVIDIA-basierten Klassengen6n_pro_win2022, gen6n_ultra_win2022gen5n_win2022, und Stream. gen4n_win2022 In dieser Umgebung unterstützt Amazon GameLift Streams Spiele und andere 3D-Anwendungen, die DirectX 11 oder DirectX 12 verwenden, sowie Game-Engines wie Unity 2022.3, Unreal Engine 4.27 und Unreal Engine 5. Streaming wird sowohl über als auch unterstützt. IPv4 IPv6

Diese Kombination aus Laufzeitumgebung und Stream-Klassen bietet eine vorhersehbare, gut unterstützte Konfiguration mit der höchsten Kompatibilität und der besten Leistung für Ihre Windows-basierten Inhalte.

Für Linux-Anwendungen

Verwenden Sie die Ubuntu 22.04 LTS-Laufzeitumgebung für Anwendungen, die für die native Ausführung unter Linux entwickelt wurden. Um die Leistung zu optimieren, wählen Sie eine der Pro- oder Ultra-Stream-Klassen (gen6n_pro, gen6n_ultragen5n_ultra, oder). gen4n_ultra Um die Kosten zu optimieren, wählen Sie eine der Stream-Klassen Small, Medium oder High (gen6n_high,,gen6n_medium, oder gen4n_high) gen6n_small gen5n_high, die Mehrmandantenfähigkeit unterstützen — eine kostengünstige Option, bei der mehrere Stream-Sitzungen gleichzeitig dieselben Rechenressourcen nutzen.

 **Important**

Die Linux-Laufzeit in Amazon GameLift Streams unterstützt kein Streaming-Over IPv6. Clients müssen Anwendungen streamen IPv4.

Kostenoptimierungen

Die Empfehlungen für den Ausgangspunkt sind zwar ein guter Anfang, Sie sollten jedoch auch andere Konfigurationsoptionen in Betracht ziehen, um die Kosten zu optimieren und gleichzeitig eine gute Leistung aufrechtzuerhalten.

Verwenden Sie die Proton-Laufzeitumgebung

Viele Windows-Anwendungen können in der Proton-Laufzeitumgebung ausgeführt werden. Proton ist eine spieloptimierte Kompatibilitätsschicht, die unter Linux läuft. Zu den Stream-Klassenoptionen für diese Laufzeit gehören leistungsstarke GPU-Ressourcen, die auf NVIDIA-Hardware laufen und DirectX 11 und, ab Proton 8.0-5, DirectX 12 unterstützen. Weitere Informationen zu dieser Option finden Sie im [Proton-Wiki](#). Wenn Sie sich dafür entscheiden, Ihre Anwendung auf Proton auszuführen, empfehlen wir Ihnen, Ihre Tests mit Proton 9.0-2 zu beginnen.

 **Important**

Proton-Laufzeiten in Amazon GameLift Streams unterstützen kein Streaming-Over. IPv6 Clients müssen Anwendungen streamen. IPv4

Important

Die Kompatibilität Ihrer Windows-Anwendung in einer Proton-Laufzeitumgebung hängt von Ihren spezifischen Anwendungsanforderungen ab. Beispielsweise bietet Proton 9.0-2 eine bessere Unterstützung als Proton 8.0-2c für Unreal Engine 5. Im Allgemeinen gilt: Je neuer Ihr Spiel ist, desto neuere Version von Proton benötigen Sie. Wir empfehlen dringend, diese Laufzeit in Ihrer lokalen Umgebung gründlich zu testen, um eine optimale Leistung sicherzustellen. Verwenden Sie unseren [Leitfaden zur Fehlerbehebung bei Proton](#), um Ihnen dabei zu helfen.

Kompilieren Sie Ihre Anwendung für Linux

Eine weitere kostensparende Option besteht darin, Ihre Anwendung so auszurichten, dass sie nativ auf Linux ausgeführt wird. Testen Sie die Anwendung zunächst auf Ihrer Seite, um sicherzustellen, dass die Linux-Version Ihrer Anwendung wie gewünscht funktioniert. Wenn Ihre Anwendung erfolgreich unter Linux ausgeführt wird, können Sie den Amazon GameLift Streams-Konfigurationsoptionen für Linux-Anwendungen folgen.

Informationen zur Cross-Compilierung von Unreal Engine-Anwendungen für Linux finden Sie im Abschnitt [Cross-Compile Toolchain](#) im Entwicklerhandbuch von Unreal Engine.

Entscheiden Sie sich für eine Konfiguration

Beachten Sie die folgenden wichtigen Fragen, um die beste Laufzeit- und Stream-Klassenkonfiguration zu ermitteln.

1. Für welche Plattform wurde Ihre Anwendung oder Ihr Spiel entwickelt? Wenn Sie eine Windows-Anwendung haben, ist die Windows-Laufzeitumgebung am einfachsten einzurichten. Wenn Ihre Anwendung für Linux entwickelt wurde, ist die Linux-Laufzeitumgebung am einfachsten. Um Kosten für das Streaming einer Windows-Anwendung zu sparen, können Sie die Proton-Laufzeitumgebung erkunden oder die Anwendung für Linux kompilieren.
2. Wie wichtig ist Leistung im Vergleich zu Kosten für Ihren Anwendungsfall? Die Windows-Laufzeitumgebung bietet zwar die beste Leistung, ihre Ausführung kann jedoch teurer sein. Im Vergleich dazu ist die Proton-Laufzeitumgebung kostengünstiger, obwohl möglicherweise etwas geringere Leistung oder potenzielle Kompatibilitätsprobleme auftreten. Dies liegt daran, dass Windows-basierte Anwendungen möglicherweise bestimmte Funktionen benötigen, die in den verfügbaren Proton-Laufzeiten noch nicht vollständig unterstützt werden. Infolgedessen

können funktionale oder grafische Unterschiede auftreten, wenn Sie Ihre Anwendung in der Proton-Umgebung ausführen. Wir empfehlen Ihnen, Ihre Anwendung in den verschiedenen Laufzeitumgebungen und Stream-Klassen zu testen, um die Kompromisse bei Leistung und Kosten zu bewerten. Eine vollständige Liste der Optionen für die Laufzeitumgebung finden Sie unter [Laufzeitumgebungen](#).

3. Was sind die grafischen Anforderungen Ihrer Anwendung? Anhand der grafischen Anforderungen Ihrer Anwendung können Sie ermitteln, welche Stream-Klassenkonfiguration am besten geeignet ist. Wenn Ihre Anwendung hohe Leistung erfordert GPUs, sollten Sie die Verwendung von Stream-Klassen mit größeren Mengen an Videospeicher (VRAM) und Systemspeicher (RAM) in Betracht ziehen. Beispielsweise bieten die Stream-Klassen gen5n und Gen6n im Vergleich zu den Gen4n-Stream-Klassen eine bis zu dreimal bessere Leistung für grafikintensive Anwendungen. Wenn Ihre Anwendung maximale GPU- und CPU-Ressourcen benötigt, sollten Sie die Stream-Klassen „Pro“ in Betracht ziehen. Umgekehrt können Sie Kosten sparen, wenn Ihre Anwendung bei geringerer grafischer Genauigkeit effektiv arbeiten kann, indem Sie Stream-Klassen verwenden, die Mehrmandantenfähigkeit unterstützen (jede der Stream-Klassen „klein“, „mittel“ oder „hoch“). Auf diese Weise können mehrere Benutzer dieselben zugrunde liegenden Hardwareressourcen gemeinsam nutzen. Eine vollständige Beschreibung der Stream-Klassenoptionen finden Sie unter [Streamen Sie Klassen](#).
4. Wie viel Aufwand sind Sie bereit, in das Setup zu investieren? Die einfachste Methode, Ihre Anwendung einzurichten, besteht darin, sie nativ mit den Windows- oder Linux-Laufzeiten auszuführen, da diese mit größerer Wahrscheinlichkeit mit Ihrer Anwendung kompatibel sind. out-of-the-box Im Gegensatz dazu erfordert die Proton-Laufzeitumgebung mehr praktische Tests, um die optimale Proton-Konfiguration für Ihre Bedürfnisse zu ermitteln. Berücksichtigen Sie bei der Entscheidung zwischen den Optionen für die Laufzeitumgebung die Zeit und die Ressourcen, die Sie dem Einrichtungs- und Testprozess zuweisen können.
5. Haben Sie Ihre Anwendung in den verschiedenen Laufzeitumgebungen und Stream-Klassen getestet? Wir empfehlen, Ihre Inhalte in verschiedenen Laufzeitumgebungen und Stream-Klassen zu testen, um zu sehen, wie sie funktionieren. Auf diese Weise können Sie anhand von Faktoren wie Stabilität, Grafikqualität, Funktionsfunktionalität und Reaktionsfähigkeit auf Eingaben die beste Lösung ermitteln.

Wie sich Ihre Konfigurationsentscheidungen auf die nächsten Schritte auswirken

Die von Ihnen gewählte Konfiguration wirkt sich direkt auf die nächsten Phasen der Einrichtung Ihrer Streaming-Umgebung aus. Das heißt:

- Erstellen einer Amazon GameLift Streams-Anwendung: Wenn Sie Ihr Spiel oder Ihre Anwendung auf Amazon GameLift Streams hochladen, müssen Sie die Laufzeitumgebung angeben, die Sie verwenden möchten. Diese Auswahl bestimmt den Typ der Stream-Gruppe, die Sie verwenden können.
- Mit einer Stream-Gruppe verknüpfen: Wenn Sie bereits über eine bestehende Stream-Gruppe verfügen, muss Ihre Wahl für die Laufzeitumgebung mit der Konfiguration dieser Gruppe übereinstimmen. Wenn Sie beispielsweise die Windows-Runtime auswählen, können Sie Ihre Anwendung nur mit einer Stream-Gruppe verknüpfen, die für Windows-Anwendungen eingerichtet wurde.
- Eine Stream-Gruppe erstellen: Wenn Sie eine neue Stream-Gruppe erstellen, müssen Sie eine Stream-Klasse auswählen, die mit der von Ihnen ausgewählten Laufzeit kompatibel ist. Die von Ihnen gewählte Stream-Klasse sollte den Grafikanforderungen und der Rechenleistung entsprechen, die Ihre Anwendung benötigt.

Wenn Sie verstehen, wie die von Ihnen ausgewählten Konfigurationseinstellungen diese nachfolgenden Schritte beeinflussen, können Sie Ihre gesamte Streaming-Implementierung besser planen und einen reibungslosen Integrationsprozess sicherstellen.

Nächste Schritte

Abhängig von der von Ihnen gewählten Konfiguration gibt es verschiedene Ansätze, mit denen Sie Ihre Anwendung für das Streaming einrichten können.

Wenn Sie die Windows- oder Linux-Runtime ausgewählt haben

Für Windows- oder Linux-Laufzeiten bestehen die nächsten Schritte darin, das Streaming in Amazon GameLift Streams einzurichten und den Stream anschließend zu testen. Weitere Informationen finden Sie unter [Ihren ersten Stream in Amazon GameLift Streams starten](#)

Wenn Sie erwägen, Proton zu verwenden

Die Kompatibilität einer Anwendung mit Proton hängt von den spezifischen Anforderungen der Anwendung ab. Daher empfehlen wir Ihnen, Ihre Anwendung auf verschiedenen Proton-Versionen zu testen, bevor Sie sie zu Amazon GameLift Streams bringen. Auf diese Weise können Sie das Proton-Setup identifizieren, das die beste Leistung und Kompatibilität für Ihre Anforderungen bietet. Durch Tests außerhalb von Amazon GameLift Streams können Sie die Leistung und Funktionalität der Anwendung überprüfen und Probleme beheben, die für die Laufzeit spezifisch sind. Weitere Informationen finden Sie unter [Kompatibilität mit Proton für Amazon GameLift Streams testen und beheben](#).

Wenn Sie eine bestimmte Proton-Konfiguration ausgewählt haben, können Sie das Streaming in Amazon GameLift Streams einrichten. Weitere Informationen finden Sie unter [Ihren ersten Stream in Amazon GameLift Streams starten](#)

Konfigurationsoptionen

Laufzeitumgebungen

Laufzeiten beziehen sich auf das zugrunde liegende Betriebssystem und die Softwareumgebung, die Ihre Anwendung auf Amazon GameLift Streams ausführt. Die wichtigsten Laufzeitoptionen sind Windows, Linux und Proton. Sie geben die Laufzeitumgebung im [Schritt 2: Konfigurieren Sie Ihre Anwendung für Amazon GameLift Streams](#) Getting Started Workflow an.

[Proton](#) ist eine Kompatibilitätsebene, mit der viele Windows-Anwendungen in einer Linux-basierten Umgebung ausgeführt werden können. Wenn Sie Proton verwenden möchten, empfehlen wir Ihnen, zu testen, wie Ihre Anwendung auf einem lokalen Computer ausgeführt wird. Weitere Informationen finden Sie unter [Kompatibilität mit Proton für Amazon GameLift Streams testen und beheben](#).

Laufzeit	Description
Microsoft Windows Server 2022-Basis	Kompatibel mit Windows-Anwendungen. Unterstützt die Verwendung von IPv4 und IPv6 in Stream-Sitzungen.
Ubuntu 22.04 LTS	Kompatibel mit Linux-Anwendungen. Unterstützt nicht die Verwendung IPv6 in Stream-Sitzungen.
Proton 9.0-2	Kompatibel mit Windows-Anwendungen. Basierend auf dem Zweig Proton experimental_9.0 . Empfohlene Version, um mit dem Testen der

Laufzeit	Description
	Kompatibilität mit Proton zu beginnen. Unterstützt die Verwendung IPv6 in Stream-Sitzungen nicht.
Proton 8.0-5	Kompatibel mit Windows-Anwendungen. Basierend auf dem Zweig Proton experimental_8.0 . Unterstützt die Verwendung IPv6 in Stream-Sitzungen nicht.
Proton 8.0-2c	Kompatibel mit Windows-Anwendungen. Basierend auf dem Zweig Proton experimental_8.0 . Unterstützt die Verwendung IPv6 in Stream-Sitzungen nicht.

Einschränkungen

Gamepad-Unterstützung ist unter Ubuntu 22.04 LTS nicht verfügbar. Andere Laufzeitumgebungen unterstützen Gamepads, abhängig vom Betriebssystem und Browser des Endbenutzers. Weitere Informationen finden Sie unter [Unterstützte Browser und Eingaben](#).

Stream-Klassen

Stream-Klassen stellen die verschiedenen Ressourcenkonfigurationen dar, die in Amazon GameLift Streams verfügbar sind und sich in CPU, GPU, RAM und anderen Spezifikationen unterscheiden. Die Stream-Klasse ist eine Konfigurationsoption einer Stream-Gruppe, die sowohl die Hardwareressourcen definiert, die einer Stream-Sitzung zugewiesen sind, als auch das Tenancy-Modell (wie viele gleichzeitige Streams auf einer einzigen virtuellen Maschine ausgeführt werden können). Sie geben die Stream-Klasse im Arbeitsablauf [Schritt 3: Verwalten Sie, wie Amazon GameLift Streams Ihre Anwendung streamt](#) „Erste Schritte“ an.

Stream-Klasse	EC2 Amazon-Konfiguration	Description
gen6n_pro _win2022	Windows-Laufzeit auf einer g6.4xlarge Amazon-Instance EC2	(NVIDIA, Pro) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität, die maximale Ressourcen benötigen. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12. Kompatibel mit Unreal Engine-Versionen bis hin zu 5.6-, 32- und 64-Bit-An-

Stream-Klasse	EC2 Amazon-Konfiguration	Description
		<p>wendungen sowie mit Anti-Cheat-Technologie. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 16. RAM: 64 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt bis zu eine gleichzeitige Stream-Sitzung.</p>
gen6n_pro	Linux-Laufzeit auf einer g6.4xlarge Amazon-Instance EC2	<p>(NVIDIA, Pro) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität, die maximale Ressourcen benötigen. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 16. RAM: 64 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt bis zu eine gleichzeitige Stream-Sitzung.</p>
gen6n_ult_ra_win2022	Windows-Laufzeit auf einer g6.2xlarge Amazon-Instance EC2	<p>(NVIDIA, Ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12. Kompatibel mit Unreal Engine-Versionen bis hin zu 5.6-, 32- und 64-Bit-Anwendungen sowie mit Anti-Cheat-Technologie. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt bis zu eine gleichzeitige Stream-Sitzung.</p>

Stream-Klasse	EC2 Amazon-Konfiguration	Description
gen6n_ultra	Linux-Laufzeit auf einer g6.2xlarge Amazon-Instance EC2	<p>(NVIDIA, Ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt bis zu eine gleichzeitige Stream-Sitzung.</p>
gen6n_high	Linux-Laufzeit auf einer g6.2xlarge EC2 Amazon-Instance mit 2:1 -Tenancy	<p>(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer 3D-Szene. moderate-to-high Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 4. RAM: 16 GB. VRAM: 12 GB.</p> <p>Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.</p>
gen6n_medium	Linux-Laufzeit auf einer g6.2xlarge EC2 Amazon-Instance mit 4:1 -Tenancy	<p>(NVIDIA, mittel) Unterstützt Anwendungen mit mäßiger Komplexität von 3D-Szenen. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 2. RAM: 8 GB. VRAM: 6 GB.</p> <p>Tenancy: Unterstützt bis zu vier gleichzeitige Stream-Sitzungen.</p>

Stream-Klasse	EC2 Amazon-Konfiguration	Description
gen6n_small	Linux-Laufzeit auf einer g6.4xlarge EC2 Amazon-Instance mit 12:1 -Tenancy	<p>(NVIDIA, klein) Unterstützt Anwendungen mit geringer 3D-Szenenkomplexität und geringer CPU-Auslastung. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 1. RAM: 4 GB. VRAM: 2 GB.</p> <p>Tenancy: Unterstützt bis zu zwölf gleichzeitige Stream-Sitzungen.</p>
gen5n_win 2022	Windows-Laufzeit auf einer g5.2xlarge Amazon-Instance EC2	<p>(NVIDIA, Ultra) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12 und DirectX 11. Unterstützt Unreal Engine bis Version 5.6, 32- und 64-Bit-Anwendungen sowie Anti-Cheat-Technologie. Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: v: 8. CPUs RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine gleichzeitige Stream-Sitzung.</p>
gen5n_ultra	Linux-Laufzeit auf einer g5.2xlarge Amazon-Instance EC2	<p>(NVIDIA, Ultra) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität. Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: v: 8. CPUs RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine gleichzeitige Stream-Sitzung.</p>

Stream-Klasse	EC2 Amazon-Konfiguration	Description
gen5n_high	Linux-Laufzeit auf einer g5.2xlarge EC2 Amazon-Instance mit 2:1 -Tenancy	<p>(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer 3D-Szene. moderate-to-high Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: v: 4. CPUs RAM: 16 GB. VRAM: 12 GB.</p> <p>Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.</p>
gen4n_win 2022	Windows-Laufzeit auf einer g4dn.2xlarge Amazon-Instance EC2	<p>(NVIDIA, Ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12 und DirectX 11. Unterstützt Unreal Engine bis Version 5.6, 32- und 64-Bit-Anwendungen sowie Anti-Cheat-Technologie. Verwendet die NVIDIA T4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 16 GB.</p> <p>Tenancy: Unterstützt eine gleichzeitige Stream-Sitzung.</p>
gen4n_ultra	Linux-Laufzeit auf einer g4dn.2xlarge Amazon-Instance EC2	<p>(NVIDIA, Ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Verwendet die NVIDIA T4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 16 GB.</p> <p>Tenancy: Unterstützt eine gleichzeitige Stream-Sitzung.</p>

Stream-Klasse	EC2 Amazon-Konfiguration	Description
gen4n_high	Linux-Laufzeit auf einer g4dn.2xlarge Amazon-Instance mit 2:1 -Tenancy EC2	(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer 3D-Szene. moderate-to-high Verwendet die NVIDIA T4 Tensor Core-GPU. Ressourcen pro Anwendung: vCPUs: 4. RAM: 16 GB. VRAM: 8 GB. Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.

Kompatibilität mit Laufzeitumgebung und Stream-Klassen

Laufzeitumgebung	Kompatible Stream-Klassen
Windows	gen6n_pro_win2022 gen6n_ultra_win2022 gen5n_win2022 gen4n_win2022
Linux (Ubuntu 22.04 LTS)	gen6n_pro gen6n_ultra gen6n_high gen6n_medium gen6n_small gen5n_ultra gen5n_high gen4n_ultra

Laufzeitumgebung	Kompatible Stream-Klassen
	gen4n_high
Proton	gen6n_pro gen6n_ultra gen6n_high gen6n_medium gen6n_small gen5n_ultra gen5n_high gen4n_ultra gen4n_high

Ihren ersten Stream in Amazon GameLift Streams starten

Dieses Tutorial führt Sie durch die ersten Schritte mit Amazon GameLift Streams, um Ihre Anwendung oder Ihr Spiel zu streamen. Amazon GameLift Streams führt Ihre Anwendung aus und streamt sie direkt in den Webbrowser Ihrer Endbenutzer. Sie erfahren, wie Sie die Anwendung, die Sie streamen möchten, hochladen und konfigurieren und wie Sie die Art und Weise verwalten, wie Amazon GameLift Streams streamt. Am Ende testen Sie, wie Ihre Anwendung auf Amazon GameLift Streams streamt, indem Sie direkt in der Amazon GameLift Streams-Konsole mit ihr interagieren.

A Bevor Sie beginnen, sollten Sie sich mit den Preisen von Amazon GameLift Streams vertraut machen.

Die Kosten für Amazon GameLift Streams finden Sie auf der [Seite mit den Preisen](#). Weitere Informationen finden Sie unter [Nutzung und Rechnungen für Amazon GameLift Streams verwalten](#).

Ihnen entstehen Kosten für die Nutzung von Amazon GameLift Streams, insbesondere wenn Sie:

- Erstellen Sie eine Amazon GameLift Streams-Anwendung in [Schritt 2: Konfigurieren Sie Ihre Anwendung für Amazon GameLift Streams](#)
- Erstellen Sie eine Stream-Gruppe in [Schritt 3: Verwalten Sie, wie Amazon GameLift Streams Ihre Anwendung streamt](#)

Überspringen Sie nicht [Schritt 5: Aufräumen \(nicht überspringen\)](#). Um unnötige Gebühren zu vermeiden, nachdem Sie Amazon GameLift Streams nicht mehr getestet haben, müssen Sie alle Ihre Ressourcen bereinigen.

Themen

- [Voraussetzungen](#)
- [Schritt 1: Laden Sie Ihre Anwendung in einen Amazon S3 S3-Bucket hoch](#)
- [Schritt 2: Konfigurieren Sie Ihre Anwendung für Amazon GameLift Streams](#)
- [Schritt 3: Verwalten Sie, wie Amazon GameLift Streams Ihre Anwendung streamt](#)
- [Schritt 4: Testen Sie Ihren Stream in Amazon GameLift Streams](#)
- [Schritt 5: Aufräumen \(nicht überspringen\)](#)

Voraussetzungen

Erledigen Sie die folgenden Aufgaben, bevor Sie mit dem Tutorial beginnen.

- Eröffnen Sie ein AWS Konto und erstellen Sie einen Benutzer mit Administratorzugriff, falls Sie noch keinen haben. Hilfe zu dieser Aufgabe finden Sie im [Einrichtung](#) Thema in diesem Handbuch. Sie müssen das Amazon GameLift Streams Web SDK zu diesem AWS CLI Zeitpunkt nicht herunterladen oder einrichten — Sie führen die folgenden Schritte mit dem durch AWS-Managementkonsole.
- Holen Sie sich eine Version Ihrer Anwendungsinhaltsdateien ohne digitale Rechteverwaltung (DRM). Sammeln Sie die für die Ausführung der Anwendung erforderlichen Dateien, einschließlich ausführbarer Dateien und Ressourcen, in einem Ordner, komprimieren Sie den Ordner jedoch nicht.

Schritt 1: Laden Sie Ihre Anwendung in einen Amazon S3 S3-Bucket hoch

Amazon GameLift Streams verwendet Amazon Simple Storage Service (Amazon S3), um Ihre Anwendungs- oder Spieldateien in der Cloud zu speichern und zum Streamen darauf zuzugreifen. In diesem Schritt laden Sie Ihre Anwendungsdateien in einen Amazon S3 S3-Bucket hoch. Führen Sie diesen Schritt in der Amazon S3 S3-Konsole aus.

Note

Die Amazon S3 S3-Speicherklasse, die Amazon GameLift Streams benötigt, ist der Standard-S3-Standard. Andere Speicherklassen wie S3 Glacier oder Objekte, die von S3 Intelligent-Tiering nach Infrequent Access oder Archive Access verschoben werden, werden von Amazon Streams nicht unterstützt. GameLift

Um die Speicherkosten zu optimieren, können Sie die Anwendung aus Ihrem S3-Bucket löschen, nachdem Sie den Vorgang abgeschlossen haben [Schritt 2: Konfigurieren Sie Ihre Anwendung für Amazon GameLift Streams](#) und sich die Anwendung im Status Bereit befindet.

Einschränkungen der Anwendung

Name	Standard	Anpas	Description
Dateien pro Anwendung	30.000 Dateien	Ja*	Die maximale Anzahl von Dateien, die Sie in einer Anwendung haben können, in diesem Konto.
Größe einer einzelnen Datei	80 GiB	Nein	Die maximale Größe einer einzelnen Datei in einer Anwendung. Beachten Sie, dass ein Gibibyte (GiB) $1024 * 1024 * 1024$ Byte entspricht.
Anwendungsgröße	100 GiB	Ja*	Die maximale Gesamtgröße einer Amazon GameLift Streams-A

Name	Standard	Anpas	Description
			nwendung in diesem Konto. Beachten Sie, dass ein Gibibyte (GiB) $1024 \times 1024 \times 1024$ Byte entspricht.

*Um eine Erhöhung zu beantragen, melden Sie sich bei [Amazon GameLift Streams](#) an AWS-Managementkonsole und öffnen Sie die Service-Kontingents-Konsole. Dort können Sie Ihre aktuellen Kontingente in der Spalte Angewendeter Kontingentwert auf Kontoebene überprüfen und eine Anfrage zur Erhöhung eines Werts einreichen.

Um Ihre Bewerbung auf Amazon S3 hochzuladen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Amazon S3 S3-Konsole unter <https://console.aws.amazon.com/s3/>.
2. Erstellen Sie einen Amazon-S3-Bucket. Geben Sie einen Bucket-Namen ein und wählen Sie einen aus AWS-Region. Diese Region muss mit der Anwendung und der Stream-Gruppe identisch sein, die Sie später erstellen werden. Eine Liste, AWS-Regionen wo Amazon GameLift Streams verfügbar ist, finden Sie unter. [AWS-Regionen und entfernte Standorte, die von Amazon GameLift Streams unterstützt werden](#) Behalten Sie für die übrigen Felder die Standardeinstellungen bei.

Weitere Anweisungen finden Sie unter [Erstellen eines Buckets](#) im Amazon Simple Storage Service-Benutzerhandbuch.

3. Öffnen Sie den neuen Bucket und laden Sie den Ordner mit Ihren Anwendungsdateien hoch.

Warning

Sie müssen Ihre Anwendungsdateien als unkomprimierten Ordner hochladen. Laden Sie keinen .zip Ordner hoch.

Warning

Stellen Sie sicher, dass es sich bei den hochgeladenen Anwendungsdateien um die richtigen handelt und dass sie innerhalb der Größenbeschränkungen für die Anwendungsdatei liegen. Wenn Sie Ihre Dateien später aktualisieren möchten, müssen Sie den Vorgang wiederholen [Schritt 2: Konfigurieren Sie Ihre Anwendung für Amazon GameLift Streams](#). Dies kann einige Minuten dauern.

Schritt 2: Konfigurieren Sie Ihre Anwendung für Amazon GameLift Streams

Was ist eine Anwendung in Amazon GameLift Streams?

Eine Amazon GameLift Streams-Anwendung ist eine Ressource, die ein Spiel oder eine interaktive Anwendung enthält, die auf der Amazon GameLift Streams-Infrastruktur läuft und Spielern Spielerlebnisse über Cloud-Streaming bietet. Die Anwendung wird auf AWS Recheninstanzen ausgeführt und rendert Spielinhalte, die über das Internet direkt auf die Geräte der Spieler gestreamt werden, sodass die Spieler das Spiel nicht mehr herunterladen, installieren oder lokal ausführen müssen.

In diesem Schritt konfigurieren Sie die Anwendung, die Sie mit Amazon GameLift Streams streamen möchten, indem Sie eine Amazon GameLift Streams-Anwendung erstellen. Wenn Sie eine Amazon GameLift Streams-Anwendung erstellen, geben Sie den Amazon S3 S3-URI für den Anwendungsordner an, den Sie in Ihren Amazon S3 S3-Bucket hochgeladen haben, und den relativen Pfad zu einer gültigen ausführbaren Datei oder Skriptdatei. Führen Sie diesen Schritt in der Amazon GameLift Streams-Konsole aus.

Um eine Amazon GameLift Streams-Anwendung mit der Amazon GameLift Streams-Konsole zu erstellen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#). Wählen Sie dasselbe aus AWS-Region wie für den Amazon S3 S3-Bucket, in den Sie Ihre Dateien hochgeladen haben. Weitere Informationen finden Sie unter [Auswahl einer Region](#) im Handbuch AWS-Managementkonsole Erste Schritte.
2. Wählen Sie in der Navigationsleiste Anwendungen und dann Anwendung erstellen aus.
3. Geben Sie in den Runtime-Einstellungen Folgendes ein:

- Laufzeitumgebung

Dies ist die Laufzeitumgebung, auf der Ihre Anwendung ausgeführt werden soll. [Amazon GameLift Streams kann entweder unter Windows, Ubuntu 22.04 LTS oder Proton ausgeführt werden.](#)

Sie können dieses Feld nach dem Abschluss der Erstellung nicht mehr bearbeiten.

Wählen Sie aus einer der folgenden Laufzeitumgebungen eigene Runtime.

- Für Linux-Anwendungen:

- Ubuntu 22.04 LTS () UBUNTU, 22_04_LTS

- Für Windows-Anwendungen:

- Microsoft Windows Server 2022-Basis (WINDOWS, 2022)
- Proton 9.0-2 () PROTON, 20250516
- Proton 8.0-5 () PROTON, 20241007
- Proton 8.0-2c () PROTON, 20230704

Lesen Sie die Beschreibungen und verwenden Sie die Vergleichscheckliste, um die optimale Laufzeitumgebung für Ihre Anwendung auszuwählen.

4. Geben Sie unter Allgemeine Einstellungen Folgendes ein:

a. Beschreibung

Dies ist eine von Menschen lesbare Bezeichnung für Ihre Anwendung. Dieser Wert muss nicht eindeutig sein. Es wird empfohlen, eine aussagekräftige Beschreibung, einen aussagekräftigen Namen oder eine aussagekräftige Bezeichnung für die Anwendung zu verwenden. Sie können dieses Feld jederzeit bearbeiten.

b. Basispfad

Dies ist der Amazon-S3-URI-Pfad zum Stammordner Ihrer Anwendung im Amazon-S3-Bucket. Der Ordner sowie jegliche Unterordner sollten die ausführbare Build-Datei und alle unterstützenden Dateien enthalten.

Ein gültiger URI ist das Bucket-Präfix, das alle Dateien enthält, die zum Ausführen und Streamen der Anwendung benötigt werden. Beispiel: Der Bucket mygamebuild enthält drei

vollständige Versionen der Build-Dateien des Spiels, jeweils in einem eigenen Ordner. Sie möchten den Build im Ordner `mygamebuild-EN101` streamen. In diesem Beispiel ist der URI `s3://amzn-s3-demo-bucket/mygamebuild-EN101`.

Sie können dieses Feld nach dem Abschluss der Erstellung nicht mehr bearbeiten.

c. Startpfad der ausführbaren Datei

Dies ist der Amazon S3 S3-URI der ausführbaren Datei, die Amazon GameLift Streams streamen wird. Die Datei muss sich im Stammordner der Anwendung befinden. Für Windows-Anwendungen muss es sich bei der Datei um eine gültige ausführbare Windows-Datei oder Batch-Datei handeln, deren Dateiname auf `.exe`, `.cmd` oder `.bat` endet. Für Linux-Anwendungen muss es sich bei der Datei um eine gültige ausführbare Linux-Binärdatei oder um ein Skript handeln, das eine erste Interpreter-Zeile enthält, die mit einem Shebang (`#!/`) beginnt.

Sie können dieses Feld nach dem Abschluss der Erstellung nicht mehr bearbeiten.

5. (Optional) Geben Sie im Anwendungsprotokollpfad Folgendes ein:

a. Pfad des Anwendungsprotokolls

Dies ist/sind der/die Pfad(e) zum Anwendungsordner oder zur Anwendungsdatei, der/die Protokolle enthält, die Sie speichern möchten. Geben Sie jeden Protokollpfad relativ zu Ihrem Anwendungsbasispfad an. Wenn Sie diese Funktion verwenden, kopiert Amazon GameLift Streams am Ende jeder Stream-Sitzung die von Ihnen angegebene (n) Datei (en) in den Amazon S3 S3-Bucket, den Sie benennen. Der Kopiervorgang wird nicht rekursiv in den Unterordnern eines Anwendungsordners ausgeführt.

Um die Protokollierung zu deaktivieren, entfernen Sie alle Anwendungsprotokollpfade und löschen Sie das Ausgabeziel für das Anwendungsprotokoll.

Sie können dieses Feld jederzeit bearbeiten.

b. Ausgabe des Anwendungsprotokolls

Dies ist der URI zum Amazon S3 S3-Bucket, in den Amazon GameLift Streams Anwendungsprotokolldateien kopiert. Dieses Feld ist erforderlich, wenn Sie einen Anwendungsprotokollpfad angeben.

Um die Protokollierung zu deaktivieren, entfernen Sie alle Anwendungsprotokollpfade und löschen Sie das Ausgabeziel für das Anwendungsprotokoll.

Sie können dieses Feld jederzeit bearbeiten.

Um Protokolldateien in Ihrem Namen zu speichern, muss Amazon GameLift Streams die Erlaubnis zum Speichern Ihres S3-Buckets erhalten. Wenn Sie Amazon GameLift Streams den Bucket für die Protokollierung erstellen lassen, wird die Berechtigungsrichtlinie bei der Erstellung automatisch angewendet. Wenn Sie Ihren eigenen Bucket bereitstellen, müssen Sie die Berechtigungsrichtlinie selbst anwenden.

Vorlage für eine Bucket-Berechtigungsrichtlinie

Kopieren Sie den folgenden Richtliniencode und wenden Sie ihn auf den Bucket an, den Sie für die Anwendungsprotokolle verwenden möchten. Achten Sie darauf, amzn-s3-demo-bucket durch den Namen des vorhandenen S3-Buckets zu ersetzen.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "PutPolicy",  
            "Effect": "Allow",  
            "Principal": {  
                "Service": [  
                    "gameliftstreams.amazonaws.com"  
                ]  
            },  
            "Action": "s3:PutObject",  
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",  
            "Condition": {  
                "StringEquals": {  
                    "aws:SourceAccount": "your 12-digit account id"  
                }  
            }  
        }  
    ]  
}
```

6. (Optional) Weisen Sie dieser Anwendung unter Tags Tags zu.

Tags sind Beschriftungen, die Ihnen bei der Organisation Ihrer AWS Ressourcen helfen können. Weitere Informationen finden Sie unter [Tagging von AWS-Ressourcen](#).

Um beispielsweise Anwendungsversionen nachzuverfolgen, verwenden Sie ein Tag wie `application-version : my-game-1121`.

7. Wählen Sie Create application aus.

Amazon GameLift Streams benötigt einige Minuten, um Ihre Bewerbung vorzubereiten. Auf der Seite Anträge befindet sich der neue Antrag im Status Bearbeitung. Wenn sich Ihre Bewerbung im Status Bereit befindet, können Sie mit dem nächsten Schritt fortfahren [Schritt 3: Verwalten Sie, wie Amazon GameLift Streams Ihre Anwendung streamt](#).

Wenn die Anfrage einen Fehler zurückgibt oder wenn die Anwendung zwar erstellt wurde, sich aber im Fehlerstatus befindet, stellen Sie sicher, dass Sie mit Benutzeranmeldedaten arbeiten, die Zugriff auf Amazon S3 und Amazon GameLift Streams beinhalten.

Note

Wenn sich eine Anwendung im Status Bereit befindet, können Sie die Anwendungsdateien in Ihrem Amazon S3 S3-Bucket sicher löschen, ohne dass sich dies auf Ihre neue Anwendung auswirkt. Dies trägt auch zur Optimierung der Speicherkosten bei. Weitere Informationen finden Sie unter [Löschen einer Anwendung](#).

Weitere Informationen finden Sie unter [Bereiten Sie eine Anwendung in Amazon GameLift Streams vor](#).

Schritt 3: Verwalten Sie, wie Amazon GameLift Streams Ihre Anwendung streamt

Was ist eine Stream-Gruppe?

Verwalten Sie mithilfe einer Stream-Gruppe, wie Amazon GameLift Streams Ihre Anwendungen streamt. Eine Stream-Gruppe ist eine Sammlung von Rechenressourcen, die Amazon GameLift Streams verwendet, um Ihre Anwendung an Endbenutzer zu streamen. Wenn Sie eine Stream-Gruppe erstellen, geben Sie die Hardwarekonfiguration (CPU, GPU, RAM) an, auf der Ihr Spiel ausgeführt werden soll (bekannt als Stream-Klasse), die geografischen Standorte, an denen Ihr Spiel ausgeführt werden kann, und die Anzahl der Streams, die gleichzeitig an jedem Standort ausgeführt werden können (bekannt als Stream-Kapazität). Sie können eine Anwendung verknüpfen, wenn Sie die Stream-Gruppe erstellen, oder bis später warten. Sie

müssen jedoch mindestens eine Anwendung verknüpfen, bevor Sie aus einer Stream-Gruppe streamen können. Nachdem eine Stream-Gruppe erstellt wurde, weist Amazon GameLift Streams Rechenressourcen an den Standorten zu, an denen Sie Stream-Kapazität zugewiesen haben. An dieser Stelle können Sie der Stream-Gruppe auch zusätzliche Anwendungen zuordnen, sodass Sie die Wahl haben, welche Sie streamen möchten.

Wenn Ihre Anwendung bereit ist, benötigen Sie als Nächstes Rechenressourcen für Amazon GameLift Streams, um sie zu streamen. In diesem Schritt verwalten Sie, wie Amazon GameLift Streams Ihre Anwendung streamt, indem Sie eine Stream-Gruppe erstellen. Führen Sie diesen Schritt in der Amazon GameLift Streams-Konsole aus.

Um eine Stream-Gruppe in der Amazon GameLift Streams-Konsole zu erstellen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#). Wählen Sie den AWS-Region Ort aus, an dem Sie Ihre Stream-Gruppe erstellen möchten. Diese Region muss mit der Region der Anwendung identisch sein, die Sie mit der Stream-Gruppe streamen möchten. Weitere Informationen finden Sie im Handbuch AWS-Managementkonsole Erste Schritte unter [Auswahl einer Region](#).
2. Um den Workflow zur Erstellung zu öffnen, wählen Sie im Navigationsbereich Stream-Gruppen und dann Stream-Gruppe erstellen aus.
3. Geben Sie unter Stream-Gruppe definieren Folgendes ein:
 - a. Beschreibung

Dies ist eine von Menschen lesbare Bezeichnung für Ihre Stream-Gruppe. Dieser Wert muss nicht eindeutig sein. Es wird empfohlen, eine aussagekräftige Beschreibung, einen aussagekräftigen Namen oder eine aussagekräftige Bezeichnung für die Stream-Gruppe zu verwenden. Sie können dieses Feld jederzeit bearbeiten.

- b. Tags
 - Tags sind Beschriftungen, die Ihnen bei der Organisation Ihrer AWS Ressourcen helfen können. Weitere Informationen finden Sie unter [Tagging von AWS-Ressourcen](#).
 4. Wählen Sie unter Stream-Klasse auswählen eine Stream-Klasse für die Stream-Gruppe aus.
 - Optionen für die Stream-Klasse
- Die Art der Rechenressourcen, mit denen Anwendungen ausgeführt und gestreamt werden sollen. Diese Wahl wirkt sich auf die Qualität des Streaming-Erlebnisses und die Kosten aus.

Sie können nur eine Stream-Klasse pro Stream-Gruppe angeben. Wählen Sie die Klasse, die am besten zu Ihrer Anwendung passt.

Stream-Klasse	Description
gen6n_pro _win2022	<p>(NVIDIA, Pro) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität, die maximale Ressourcen benötigen. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12. Kompatibel mit Unreal Engine-Versionen bis hin zu 5.6-, 32- und 64-Bit-Anwendungen sowie mit Anti-Cheat-Technologie. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 16. RAM: 64 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt bis zu eine gleichzeitige Stream-Sitzung.</p>
gen6n_pro	<p>(NVIDIA, Pro) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität, die maximale Ressourcen benötigen. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 16. RAM: 64 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt bis zu eine gleichzeitige Stream-Sitzung.</p>
gen6n_ult ra_win2022	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12. Kompatibel mit Unreal Engine-Versionen bis hin zu 5.6-, 32- und 64-Bit-Anwendungen sowie mit Anti-Cheat-Technologie. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt bis zu eine gleichzeitige Stream-Sitzung.</p>

Stream-Klasse	Description
gen6n_ultra	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt bis zu eine gleichzeitige Stream-Sitzung.</p>
gen6n_high	<p>(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer moderate-to-high 3D-Szene. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 4. RAM: 16 GB. VRAM: 12 GB.</p> <p>Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.</p>
gen6n_medium	<p>(NVIDIA, mittel) Unterstützt Anwendungen mit moderater 3D-Szenenkomplexität. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 2. RAM: 8 GB. VRAM: 6 GB.</p> <p>Tenancy: Unterstützt bis zu vier gleichzeitige Stream-Sitzungen.</p>
gen6n_small	<p>(NVIDIA, klein) Unterstützt Anwendungen mit geringer 3D-Szenenkomplexität und geringer CPU-Auslastung. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 1. RAM: 4 GB. VRAM: 2 GB.</p> <p>Tenancy: Unterstützt bis zu zwölf gleichzeitige Stream-Sitzungen.</p>

Stream-Klasse	Description
gen5n_win2022	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12 und DirectX 11. Unterstützt Unreal Engine bis Version 5.6, 32- und 64-Bit-Anwendungen sowie Anti-Cheat-Technologie. Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: v: 8. CPUs RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine gleichzeitige Stream-Sitzung.</p>
gen5n_high	<p>(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer moderate-to-high 3D-Szene. Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: v: 4. CPUs RAM: 16 GB. VRAM: 12 GB.</p> <p>Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.</p>
gen5n_ultra	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität. Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: v: 8. CPUs RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine gleichzeitige Stream-Sitzung.</p>

Stream-Klasse	Description
gen4n_win2022	<p>(NVIDIA, Ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12 und DirectX 11. Unterstützt Unreal Engine bis Version 5.6, 32- und 64-Bit-Anwendungen sowie Anti-Cheat-Technologie. Verwendet die NVIDIA T4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 16 GB.</p> <p>Tenancy: Unterstützt eine gleichzeitige Stream-Sitzung.</p>
gen4n_high	<p>(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer moderate-to-high 3D-Szene. Verwendet die NVIDIA T4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 4. RAM: 16 GB. VRAM: 8 GB.</p> <p>Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.</p>
gen4n_ultra	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Verwendet die NVIDIA T4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 16 GB.</p> <p>Tenancy: Unterstützt eine gleichzeitige Stream-Sitzung.</p>

Wählen Sie Next, um fortzufahren.

5. Wählen Sie unter Anwendung verknüpfen eine Anwendung aus, die Sie streamen möchten, oder wählen Sie "Keine Anwendung", um zu einem späteren Zeitpunkt eine Anwendung auszuwählen. Sie können die Stream-Gruppe bearbeiten, nachdem sie erstellt wurde, um Anwendungen hinzuzufügen oder zu entfernen. Sie können nur eine Anwendung verknüpfen, die sich im Ready Status befindet und deren Laufzeit mit der von Ihnen ausgewählten Stream-Klasse kompatibel ist. Standardmäßig sind dies die einzigen Anwendungen, die in der Tabelle angezeigt werden.

Um alle Bewerbungen im Ready Status zu sehen, wählen Sie All runtimes in der Drop-down-Liste aus.

 Note

Wenn Ihre Bewerbung nicht aufgeführt ist, überprüfen Sie die aktuelle AWS-Region Einstellung. Sie können eine Anwendung nur mit einer Stream-Gruppe verknüpfen, die sich in derselben Region befindet.

Wählen Sie Next, um fortzufahren.

6. Wählen Sie unter Stream-Einstellungen konfigurieren unter Standorte und Kapazität einen oder mehrere Standorte aus, an denen Ihre Stream-Gruppe die Kapazität zum Streamen Ihrer Anwendung haben soll. Standardmäßig wurde die Region, in der Sie die Stream-Gruppe erstellen, der so genannte primäre Standort, bereits zu Ihrer Stream-Gruppe hinzugefügt und kann nicht entfernt werden. Du kannst weitere Standorte hinzufügen, indem du das Kästchen neben jedem Standort anklickst, den du hinzufügen möchtest. Für eine geringere Latenz und eine bessere Streaming-Qualität sollten Sie Standorte wählen, die sich näher an Ihren Benutzern befinden.

Für jeden Standort können Sie seine Streaming-Kapazität angeben. Die Stream-Kapazität ist die Anzahl der gleichzeitigen Streams, die zu einem bestimmten Zeitpunkt aktiv sein können. Sie legen die Stream-Kapazität pro Speicherort in jeder Stream-Gruppe fest.

- Always-On-Kapazität: Wenn diese Einstellung ungleich Null ist, gibt sie die minimale Streaming-Kapazität an, die Ihnen zugewiesen wird und die niemals wieder an den Dienst zurückgegeben wird. Sie zahlen jederzeit für diese Basiskapazität, unabhängig davon, ob sie genutzt oder nicht genutzt wird.
- Maximale Kapazität: Dies gibt die maximale Kapazität an, die der Service Ihnen zuweisen kann. Es kann einige Minuten dauern, bis neu erstellte Streams gestartet werden. Im Leerlauf wird Kapazität wieder an den Dienst zurückgegeben. Sie zahlen für die Ihnen zugewiesene Kapazität, bis sie freigegeben wird.
- Zielkapazität im Leerlauf: Dies gibt ungenutzte Kapazität an, die der Service Ihnen im Voraus zuweist und für Sie im Vorgriff auf zukünftige Aktivitäten bereithält. Dies trägt dazu bei, Ihre Benutzer vor Verzögerungen bei der Kapazitätszuweisung zu schützen. Sie zahlen für Kapazität, die in diesem vorsätzlichen Ruhezustand gehalten wird.

Sie können Ihre gesamte Stream-Kapazität jederzeit erhöhen oder verringern, um der sich ändernden Nutzernachfrage nach einem Standort gerecht zu werden, indem Sie eine der Kapazitäten anpassen. Amazon GameLift Streams erfüllt Streaming-Anfragen mithilfe der inaktiven, vorab zugewiesenen Ressourcen im Always-On-Kapazitätspool, sofern welche verfügbar sind. Wenn die gesamte Always-On-Kapazität genutzt wird, stellt Amazon GameLift Streams zusätzliche Rechenressourcen bis zu der in On-Demand-Kapazität angegebenen Höchstzahl bereit. Wenn die zugewiesene Kapazität skaliert wird, spiegelt sich die Änderung in Ihren Gesamtkosten für die Stream-Gruppe wider.

Verknüpfte Anwendungen werden automatisch an jeden aktivierten Standort repliziert. Eine Anwendung muss die Replikation an einem Remote-Standort abgeschlossen haben, bevor der Remote-Standort einen Stream hosten kann. Um den Replikationsstatus zu überprüfen, öffnen Sie die Stream-Gruppe, nachdem sie erstellt wurde, und schlagen Sie in der Tabelle der verknüpften Anwendungen in der Spalte Replikationsstatus nach. Klicken Sie auf den aktuellen Status, um den Replikationsstatus für jeden hinzugefügten Standort zu sehen.

 Note

Anwendungsdaten werden an allen aktivierten Speicherorten gespeichert, einschließlich des primären Speicherorts für diese Stream-Gruppe. Stream-Sitzungsdaten werden sowohl am primären Standort als auch an dem Ort gespeichert, an dem das Streaming stattgefunden hat.

7. Überprüfen Sie unter Stream-Gruppe überprüfen und erstellen die Konfiguration Ihrer Stream-Gruppe und nehmen Sie bei Bedarf Änderungen vor. Wenn alles korrekt ist, wähle Stream-Gruppe erstellen.

Weitere Informationen finden Sie unter [Streaming mit einer Amazon GameLift Streams-Stream-Gruppe verwalten](#).

Schritt 4: Testen Sie Ihren Stream in Amazon GameLift Streams

Was ist eine Stream-Sitzung?

Bezieht sich auf den Stream selbst. Dies ist eine Instanz eines Streams, den Amazon GameLift Streams vom Server an den Endbenutzer überträgt. Eine Stream-Sitzung läuft auf einer

Rechenressource oder Stream-Kapazität, die von einer Stream-Gruppe zugewiesen wurde. Wird auch kurz als Stream bezeichnet.

Sie können sehen, wie Ihre Anwendung streamt, indem Sie sie direkt in der Amazon GameLift Streams-Konsole ausführen. Wenn Sie einen Stream starten, verwendet Amazon GameLift Streams eine der Rechenressourcen, die Ihre Stream-Gruppe zuweist. Sie müssen also über verfügbare Kapazität in Ihrer Stream-Gruppe verfügen.

Um Ihren Stream in der Amazon GameLift Streams-Konsole zu testen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Sie können einen Stream auf verschiedene Arten testen. Beginne auf der Seite Stream-Gruppen oder Stream testen und folge diesen Schritten:
 - a. Wählen Sie eine Stream-Gruppe aus, die Sie zum Streamen verwenden möchten.
 - b. Wenn du auf der Seite Stream-Gruppen startest, wähle Stream testen. Wenn du auf der Seite „Stream testen“ startest, wähle Wählen aus. Dadurch wird die Seite „Stream-Konfiguration testen“ für die ausgewählte Stream-Gruppe geöffnet.
 - c. Wählen Sie unter Verknüpfte Anwendungen eine Anwendung aus.
 - d. Wählen Sie unter Standort einen Standort mit verfügbarer Kapazität aus.
 - e. (Optional) Geben Sie unter Programmkonfigurationen Befehlszeilenargumente oder Umgebungsvariablen ein, die beim Start an die Anwendung übergeben werden sollen.
 - f. Bestätigen Sie Ihre Auswahl und wählen Sie Stream testen aus.
3. Nachdem dein Stream geladen wurde, kannst du die folgenden Aktionen in deinem Stream ausführen:
 - a. Um Eingaben wie Maus, Tastatur und Gamepad zu verbinden (mit Ausnahme von Mikrofonen, die im Teststream nicht unterstützt werden), wähle Eingabe anhängen. Du verbindest deine Maus automatisch, wenn du den Cursor in das Stream-Fenster bewegst.
 - b. Um Dateien, die während der Streaming-Sitzung erstellt wurden, am Ende der Sitzung in einen Amazon S3 S3-Bucket exportieren zu lassen, wählen Sie Dateien exportieren und geben Sie die Bucket-Details an. Exportierte Dateien finden Sie auf der Seite Sessions.
 - c. Um den Stream im Vollbildmodus anzusehen, wählen Sie Vollbild. Drücken Sie Esc, um diese Aktion rückgängig zu machen.

- Um den Stream zu beenden, wählen Sie Sitzung beenden. Wenn die Verbindung zum Stream unterbrochen wird, steht die Stream-Kapazität zum Starten eines weiteren Streams zur Verfügung.

 Note

Die Funktion Stream testen in der Amazon GameLift Streams-Konsole unterstützt keine Mikrofone.

Schritt 5: Aufräumen (nicht überspringen)

 **Vermeiden Sie unnötige Kosten**

Einer Stream-Gruppe fallen Kosten an, wenn ihr Kapazität zugewiesen wurde, auch wenn diese Kapazität ungenutzt ist. Um unnötige Kosten zu vermeiden, skalieren Sie die Kapazitäten Ihrer Stream-Gruppe auf die von Ihnen benötigte Größe. Wir empfehlen Ihnen, Ihre Always-On-Kapazität während der Entwicklung auf Null zu skalieren, wenn sie nicht genutzt wird. Weitere Informationen finden Sie unter [Bewährte Methoden zur Verwaltung der Amazon GameLift Streams-Kosten](#).

Nachdem Sie das Tutorial abgeschlossen haben und Ihre Anwendung nicht mehr streamen müssen, gehen Sie wie folgt vor, um Ihre Amazon GameLift Streams-Ressourcen zu bereinigen.

Um eine Stream-Gruppe mit der Amazon GameLift Streams-Konsole zu löschen

- Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
- Um eine Liste Ihrer vorhandenen Stream-Gruppen anzuzeigen, wählen Sie im Navigationsbereich Stream-Gruppen aus.
- Wählen Sie den Namen der Stream-Gruppe aus, die Sie löschen möchten.
- Wählen Sie auf der Detailseite der Stream-Gruppe die Option Löschen aus.
- Bestätigen Sie im Dialogfeld „Löschen“ die Löschaktion.

Amazon GameLift Streams beginnt mit der Freigabe von Rechenressourcen und dem Löschen der Stream-Gruppe. Während dieser Zeit befindet sich die Stream-Gruppe im Status Löschen. Nachdem Amazon GameLift Streams die Stream-Gruppe gelöscht hat, können Sie sie nicht mehr abrufen.

Um eine Anwendung mithilfe der Amazon GameLift Streams-Konsole zu löschen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Wählen Sie in der Navigationsleiste Anwendungen aus, um eine Liste Ihrer vorhandenen Anwendungen anzuzeigen. Wählen Sie die Anwendung aus, die Sie löschen möchten.
3. Wählen Sie auf der Detailseite der Anwendung die Option Löschen aus.
4. Bestätigen Sie im Dialogfeld Löschen die Löschaktion.

Amazon GameLift Streams beginnt mit dem Löschen der Anwendung. Während dieser Zeit befindet sich die Anwendung im Deleting Status. Nachdem Amazon GameLift Streams die Anwendung gelöscht hat, können Sie sie nicht mehr abrufen.

Weitere Informationen finden Sie unter [Löschen Sie eine Stream-Gruppe](#) und [Löschen einer Anwendung](#).

Verwaltung Ihrer Streams mit Amazon GameLift Streams

Dieser Abschnitt enthält detaillierte Informationen zum Streamen mit Amazon GameLift Streams. Erfahren Sie mehr über die Streaming-Ressourcen (eine Anwendung und eine Stream-Gruppe), die Eigenschaften zur Skalierung Ihres Streamings (Stream-Kapazität und Standorte) und den Stream selbst (eine Stream-Sitzung). Sie können alle Aufgaben erledigen, die für die Einrichtung von Streaming mit Amazon GameLift Streams erforderlich sind, indem Sie die Amazon GameLift Streams-Konsole oder die Amazon GameLift Streams-CLI-Befehle verwenden.

Wenn Sie Amazon GameLift Streams zum ersten Mal verwenden, finden Sie weitere Informationen unter [Ihren ersten Stream in Amazon GameLift Streams starten](#), die Sie durch den gesamten Workflow führen.

Themen

- [Die wichtigsten Konzepte](#)
- [Bereiten Sie eine Anwendung in Amazon GameLift Streams vor](#)
- [Streaming mit einer Amazon GameLift Streams-Stream-Gruppe verwalten](#)
- [Übersicht über Stream-Gruppen für mehrere Anwendungen](#)
- [Stream-Sitzungen mit Amazon GameLift Streams starten](#)
- [Stream-Sitzungsdateien exportieren](#)

Die wichtigsten Konzepte

Anwendung

Eine Amazon GameLift Streams-Anwendung ist eine Ressource, die ein Spiel oder eine interaktive Anwendung enthält, die auf der Amazon GameLift Streams-Infrastruktur läuft und Spielern Spielerlebnisse über Cloud-Streaming bietet. Die Anwendung wird auf AWS Recheninstanzen ausgeführt und rendert Spielinhalte, die über das Internet direkt auf die Geräte der Spieler gestreamt werden, sodass die Spieler das Spiel nicht mehr herunterladen, installieren oder lokal ausführen müssen.

Stream-Gruppen für mehrere Anwendungen

Eine Stream-Gruppe, die mit mehreren Anwendungen verknüpft ist. Diese many-to-one Beziehung ermöglicht es Ihnen, mehrere Anwendungen zu streamen, indem Sie dieselbe Konfiguration verwenden, die Sie in einer einzigen Stream-Gruppe eingerichtet haben. Wenn

Sie eine Stream-Sitzung starten, geben Sie alle verknüpften Anwendungen an. Anschließend GameLift streamt Amazon Streams diese Anwendung mithilfe der verfügbaren Stream-Kapazität in dieser Stream-Gruppe.

Stream-Gruppen mit mehreren Standorten

Eine Stream-Gruppe, die so konfiguriert ist, dass sie neben dem primären Standort (dem, an AWS-Region dem Sie die Streamgruppe erstellt haben) auch Anwendungen und Stream-Sitzungen von mehreren Standorten aus hostet. Sie verwalten die Kapazität für jeden Standort.

Mehrmandantenfähigkeit

Tenancy bezieht sich darauf, wie viele gleichzeitige Streams von einer einzigen Rechenressource in Amazon GameLift Streams unterstützt werden können. Multi-Tenancy ist eine Funktion, die es mehreren Benutzern ermöglicht, dieselben zugrunde liegenden Hardwareressourcen gemeinsam zu nutzen. Dies kann eine kostengünstige Option für Anwendungen sein, die nicht die maximale Hardwarekapazität erfordern. Eine Stream-Klasse mit Mehrmandantenfähigkeit kann mehrere Streams zum Preis einer Ressource hosten. Stream-Klassen „High“ unterstützen Multi-Tenancy, sodass zwei Anwendungen gleichzeitig auf einer einzigen Rechenressource ausgeführt werden können, während Stream-Klassen „Ultra“ -Multi-Tenancy nicht unterstützen.

Stream-Gruppe

Verwalten Sie mithilfe einer Stream-Gruppe, wie Amazon GameLift Streams Ihre Anwendungen streamt. Eine Stream-Gruppe ist eine Sammlung von Rechenressourcen, die Amazon GameLift Streams verwendet, um Ihre Anwendung an Endbenutzer zu streamen. Wenn Sie eine Stream-Gruppe erstellen, geben Sie die Hardwarekonfiguration (CPU, GPU, RAM) an, auf der Ihr Spiel ausgeführt werden soll (bekannt als Stream-Klasse), die geografischen Standorte, an denen Ihr Spiel ausgeführt werden kann, und die Anzahl der Streams, die gleichzeitig an jedem Standort ausgeführt werden können (bekannt als Stream-Kapazität). Sie können eine Anwendung verknüpfen, wenn Sie die Stream-Gruppe erstellen, oder bis später warten. Sie müssen jedoch mindestens eine Anwendung verknüpfen, bevor Sie aus einer Stream-Gruppe streamen können. Nachdem eine Stream-Gruppe erstellt wurde, weist Amazon GameLift Streams Rechenressourcen an den Standorten zu, an denen Sie Stream-Kapazität zugewiesen haben. An dieser Stelle können Sie der Stream-Gruppe auch zusätzliche Anwendungen zuordnen, sodass Sie die Wahl haben, welche Sie streamen möchten.

Stream-Kapazität

Stellt die Anzahl gleichzeitiger Streams dar, die gleichzeitig aktiv sein können. Sie legen die Stream-Kapazität pro Speicherort in jeder Stream-Gruppe fest. Sie konfigurieren Always-On-Kapazität und maximale Kapazität. Die maximale Kapazität entspricht der Gesamtzahl der

möglichen Streams, wobei Always-On-Kapazität mit zusätzlicher Kapazität kombiniert wird, die nach Bedarf bereitgestellt wird.

Stream-Sitzung

Bezieht sich auf den Stream selbst. Dies ist eine Instanz eines Streams, den Amazon GameLift Streams vom Server an den Endbenutzer überträgt. Eine Stream-Sitzung läuft auf einer Rechenressource oder Stream-Kapazität, die von einer Stream-Gruppe zugewiesen wurde. Wird auch kurz als Stream bezeichnet.

Bereiten Sie eine Anwendung in Amazon GameLift Streams vor

Um Streaming mit Amazon GameLift Streams einzurichten, laden Sie zuerst das Spiel oder eine andere Anwendung hoch, die Sie streamen möchten, und konfigurieren dann eine Anwendungsressource in Amazon GameLift Streams, um Metadaten zu Ihrem Spiel zu definieren. Eine Amazon GameLift Streams-Anwendung besteht aus den von Ihnen hochgeladenen Dateien (ausführbare Dateien und alle unterstützenden Dateien) und einer Konfiguration, die Amazon GameLift Streams anweist, welche ausführbare Datei beim Streaming ausgeführt werden soll.

Jede Amazon GameLift Streams-Anwendung stellt eine einzelne Version Ihrer Inhalte dar. Wenn Sie mehrere Versionen haben, müssen Sie für jede Version eine separate Anwendung erstellen. Nachdem Sie eine Anwendung erstellt haben, können Sie die Dateien nicht aktualisieren. Wenn Sie die ausführbare Datei oder unterstützende Dateien aktualisieren müssen, müssen Sie eine neue Amazon GameLift Streams-Anwendung erstellen.

Bevor Sie hochladen

Bevor Sie eine Amazon GameLift Streams-Anwendung erstellen, stellen Sie sicher, dass Ihr Spiel die folgenden Einschränkungen einhält.

Name	Standard	Anpas	Description
Dateien pro Anwendung	30.000 Dateien	Ja*	Die maximale Anzahl von Dateien, die Sie in einer Anwendung haben können, in diesem Konto.
Größe einer einzelnen Datei	80 GiB	Nein	Die maximale Größe einer einzelnen Datei

Name	Standard	Anpas	Description
			in einer Anwendung. Beachten Sie, dass ein Gibibyte (GiB) 1024^*1024^*1024 Byte entspricht.
Anwendungsgröße	100 GiB	Ja*	Die maximale Gesamtgröße einer Amazon GameLift Streams-Anwendung in diesem Konto. Beachten Sie, dass ein Gibibyte (GiB) 1024^*1024^*1024 Byte entspricht.

*Um eine Erhöhung zu beantragen, melden Sie sich bei [Amazon GameLift Streams](#) an AWS-Managementkonsole und öffnen Sie die Service-Kontingents-Konsole. Dort können Sie Ihre aktuellen Kontingente in der Spalte Angewandter Kontingentwert auf Kontoebene überprüfen und eine Anfrage zur Erhöhung eines Werts einreichen.

Note

Um Zeit und Mühe zu sparen, stellen Sie sicher, dass es sich bei den Dateien, die Sie hochladen möchten, um die richtige Version Ihrer Anwendung handelt. Sie können zwar später neue Versionen hochladen, müssen den [Erstellen einer Anwendung](#) Schritt jedoch für jede Version wiederholen.

Laden Sie Ihre Anwendung in einen Amazon S3 S3-Bucket hoch

Nachdem Sie Ihr Spiel für Amazon GameLift Streams vorbereitet haben, ist es an der Zeit, es in einen Amazon Simple Storage Service (Amazon S3) -Bucket in Ihrem AWS Konto hochzuladen.

Note

Die Amazon S3 S3-Speicherklasse, die Amazon GameLift Streams benötigt, ist der Standard-S3-Standard. Andere Speicherklassen wie S3 Glacier oder Objekte, die von S3 Intelligent-

Tiering nach Infrequent Access oder Archive Access verschoben werden, werden von Amazon Streams nicht unterstützt. GameLift

Um die Speicherkosten zu optimieren, können Sie die Anwendung aus Ihrem S3-Bucket löschen, nachdem Sie den Vorgang abgeschlossen haben [Erstellen einer Anwendung](#) und sich die Anwendung im Status Bereit befindet.

Um Ihre Bewerbung auf Amazon S3 hochzuladen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Amazon S3 S3-Konsole unter <https://console.aws.amazon.com/s3/>.
2. Erstellen Sie einen Amazon-S3-Bucket. Geben Sie einen Bucket-Namen ein und wählen Sie einen aus AWS-Region. Diese Region muss mit der Anwendung und der Stream-Gruppe identisch sein, die Sie später erstellen werden. Eine Liste, AWS-Regionen wo Amazon GameLift Streams verfügbar ist, finden Sie unter. [AWS-Regionen und entfernte Standorte, die von Amazon GameLift Streams unterstützt werden](#) Behalten Sie für die übrigen Felder die Standardeinstellungen bei.

Weitere Anweisungen finden Sie unter [Bucket erstellen](#) im Amazon Simple Storage Service-Benutzerhandbuch.

3. Öffnen Sie den neuen Bucket und laden Sie den Ordner mit Ihren Anwendungsdateien hoch.

 **Warning**

Sie müssen Ihre Anwendungsdateien als unkomprimierten Ordner hochladen. Laden Sie keinen .zip Ordner hoch.

Erstellen einer Anwendung

Eine Amazon GameLift Streams-Anwendung ist eine Ressource, die ein Spiel oder eine interaktive Anwendung enthält, die auf der Amazon GameLift Streams-Infrastruktur läuft und Spielern Spielerlebnisse über Cloud-Streaming bietet. Die Anwendung wird auf AWS Recheninstanzen ausgeführt und rendert Spielinhalte, die über das Internet direkt auf die Geräte der Spieler gestreamt werden, sodass die Spieler das Spiel nicht mehr herunterladen, installieren oder lokal ausführen müssen.

Wenn Sie eine Amazon GameLift Streams-Anwendung erstellen, geben Sie den Amazon S3 S3-URI für den Anwendungsordner an, den Sie in Ihren Amazon S3 S3-Bucket hochgeladen haben, und den relativen Pfad zu einer gültigen ausführbaren Datei oder Skriptdatei.

Amazon GameLift Streams synchronisiert Ihre Anwendungsdateien nicht mit den Dateien im Amazon S3 S3-Bucket. Wenn Sie die Dateien in Ihrer Amazon GameLift Streams-Anwendung aktualisieren möchten, müssen Sie eine neue Amazon GameLift Streams-Anwendung erstellen.

Console

Um eine Amazon GameLift Streams-Anwendung mit der Amazon GameLift Streams-Konsole zu erstellen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#). Wählen Sie dasselbe aus AWS-Region wie für den Amazon S3 S3-Bucket, in den Sie Ihre Dateien hochgeladen haben. Weitere Informationen finden Sie unter [Auswahl einer Region](#) im Handbuch AWS-Managementkonsole Erste Schritte.
2. Wählen Sie in der Navigationsleiste Anwendungen und dann Anwendung erstellen aus.
3. Geben Sie in den Runtime-Einstellungen Folgendes ein:
 - Laufzeitumgebung

Dies ist die Laufzeitumgebung, auf der Ihre Anwendung ausgeführt werden soll. [Amazon GameLift Streams kann entweder unter Windows, Ubuntu 22.04 LTS oder Proton ausgeführt werden.](#)

Sie können dieses Feld nach dem Abschluss der Erstellung nicht mehr bearbeiten.

Wählen Sie aus einer der folgenden Laufzeitumgebungen eigene Runtime.

- Für Linux-Anwendungen:
 - Ubuntu 22.04 LTS () UBUNTU, 22_04_LTS
- Für Windows-Anwendungen:
 - Microsoft Windows Server 2022-Basis (WINDOWS, 2022)
 - Proton 9.0-2 () PROTON, 20250516
 - Proton 8.0-5 () PROTON, 20241007
 - Proton 8.0-2c () PROTON, 20230704

Lesen Sie die Beschreibungen und verwenden Sie die Vergleichscheckliste, um die optimale Laufzeitumgebung für Ihre Anwendung auszuwählen.

4. Geben Sie unter Allgemeine Einstellungen Folgendes ein:

a. Beschreibung

Dies ist eine von Menschen lesbare Bezeichnung für Ihre Anwendung. Dieser Wert muss nicht eindeutig sein. Es wird empfohlen, eine aussagekräftige Beschreibung, einen aussagekräftigen Namen oder eine aussagekräftige Bezeichnung für die Anwendung zu verwenden. Sie können dieses Feld jederzeit bearbeiten.

b. Basispfad

Dies ist der Amazon-S3-URI-Pfad zum Stammordner Ihrer Anwendung im Amazon-S3-Bucket. Der Ordner sowie jegliche Unterordner sollten die ausführbare Build-Datei und alle unterstützenden Dateien enthalten.

Ein gültiger URI ist das Bucket-Präfix, das alle Dateien enthält, die zum Ausführen und Streamen der Anwendung benötigt werden. Beispiel: Ein Bucket namens mygamebuild enthält drei vollständige Versionen der Build-Dateien des Spiels, jede in einem separaten Ordner. Sie möchten den Build im Ordner mygamebuild-EN101 streamen. In diesem Beispiel ist der URI s3://amzn-s3-demo-bucket/mygamebuild-EN101.

Sie können dieses Feld nach dem Abschluss der Erstellung nicht mehr bearbeiten.

c. Startpfad der ausführbaren Datei

Dies ist der Amazon S3 S3-URI der ausführbaren Datei, die Amazon GameLift Streams streamen wird. Die Datei muss sich im Stammordner der Anwendung befinden. Für Windows-Anwendungen muss es sich bei der Datei um eine gültige ausführbare Windows-Datei oder Batchdatei handeln, deren Dateiname auf .exe, .cmd oder .bat endet. Für Linux-Anwendungen muss es sich bei der Datei um eine gültige ausführbare Linux-Binärdatei oder um ein Skript handeln, das eine erste Interpreterzeile enthält, die mit einem Shebang ("") beginnt. #!

Sie können dieses Feld nach dem Abschluss der Erstellung nicht mehr bearbeiten.

5. (Optional) Geben Sie im Feld Pfad zum Anwendungsprotokoll Folgendes ein:

a. Pfad des Anwendungsprotokolls

Dies ist/sind der/die Pfad(e) zum Anwendungsordner oder zur Anwendungsdatei, der/die Protokolle enthält, die Sie speichern möchten. Geben Sie jeden Protokollpfad relativ zu Ihrem Anwendungsbasispfad an. Wenn Sie diese Funktion verwenden, kopiert Amazon GameLift Streams am Ende jeder Stream-Sitzung die von Ihnen angegebene (n) Datei (en) in den Amazon S3 S3-Bucket, den Sie benennen. Der Kopievorgang wird nicht rekursiv in den Unterordnern eines Anwendungsordners ausgeführt.

Um die Protokollierung zu deaktivieren, entfernen Sie alle Anwendungsprotokollpfade und löschen Sie das Ausgabeziel für das Anwendungsprotokoll.

Sie können dieses Feld jederzeit bearbeiten.

b. Ausgabe des Anwendungsprotokolls

Dies ist der URI zum Amazon S3 S3-Bucket, in den Amazon GameLift Streams Anwendungsprotokolldateien kopiert. Dieses Feld ist erforderlich, wenn Sie einen Anwendungsprotokollpfad angeben.

Um die Protokollierung zu deaktivieren, entfernen Sie alle Anwendungsprotokollpfade und löschen Sie das Ausgabeziel für das Anwendungsprotokoll.

Sie können dieses Feld jederzeit bearbeiten.

Um Protokolldateien in Ihrem Namen zu speichern, muss Amazon GameLift Streams die Erlaubnis zum Speichern Ihres S3-Buckets erhalten. Wenn Sie Amazon GameLift Streams den Bucket für die Protokollierung erstellen lassen, wird die Berechtigungsrichtlinie bei der Erstellung automatisch angewendet. Wenn Sie Ihren eigenen Bucket bereitstellen, müssen Sie die Berechtigungsrichtlinie selbst anwenden.

Vorlage für eine Bucket-Berechtigungsrichtlinie

Kopieren Sie den folgenden Richtliniencode und wenden Sie ihn auf den Bucket an, den Sie für die Anwendungsprotokolle verwenden möchten. Achten Sie darauf, amzn-s3-demo-bucket durch den Namen des vorhandenen S3-Buckets zu ersetzen.

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "PutPolicy",  
            "Effect": "Allow",  
            "Action": "s3:PutObject",  
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"  
        }  
    ]  
}
```

```

    "Effect": "Allow",
    "Principal": {
        "Service": [
            "gameliftstreams.amazonaws.com"
        ]
    },
    "Action": "s3:PutObject",
    "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "your 12-digit account id"
        }
    }
}
]
}

```

6. (Optional) Weisen Sie dieser Anwendung unter Tags Tags zu.

Tags sind Beschriftungen, die Ihnen bei der Organisation Ihrer AWS Ressourcen helfen können. Weitere Informationen finden Sie unter [Taggen Ihrer AWS-Ressourcen](#).

Um beispielsweise Anwendungsversionen nachzuverfolgen, verwenden Sie ein Tag wie `application-version : my-game-1121`.

7. Wählen Sie Create application aus.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren AWS-Region. Anweisungen zur Einrichtung finden Sie unter [Laden Sie das herunter AWS CLI](#).

Um eine Anwendung mit dem zu erstellen AWS CLI

AWS CLI Verwenden Sie dabei den [CreateApplication](#)Befehl, der an Ihren Inhalt angepasst ist.

```

aws gameliftstreams create-application \
--description "MyGame v1" \
--runtime-environment '{"Type":"PROTON", "Version":"20241007"}' \
--executable-path "launcher.exe" \

```

```
--application-source-uri "s3://amzn-s3-demo-bucket/example"
```

where

- **description:**

Dies ist eine von Menschen lesbare Bezeichnung für Ihre Anwendung. Dieser Wert muss nicht eindeutig sein. Es wird empfohlen, eine aussagekräftige Beschreibung, einen aussagekräftigen Namen oder eine aussagekräftige Bezeichnung für die Anwendung zu verwenden. Sie können dieses Feld jederzeit bearbeiten.

- **runtime-environment:**

Dies ist die Laufzeitumgebung, auf der Ihre Anwendung ausgeführt werden soll. [Amazon GameLift Streams kann entweder unter Windows, Ubuntu 22.04 LTS oder Proton ausgeführt werden.](#)

Sie können dieses Feld nach dem Abschluss der Erstellung nicht mehr bearbeiten.

Wählen Sie aus einer der folgenden Laufzeitumgebungen eigene Runtime.

- Für Linux-Anwendungen:

- Ubuntu 22.04 LTS () Type=UBUNTU, Version=22_04_LTS

- Für Windows-Anwendungen:

- Microsoft Windows Server 2022-Basis (Type=WINDOWS, Version=2022)
 - Proton 9.0-2 () Type=PROTON, Version=20250516
 - Proton 8.0-5 () Type=PROTON, Version=20241007
 - Proton 8.0-2c () Type=PROTON, Version=20230704

- **application-source-uri:**

Dies ist der Amazon-S3-URI-Pfad zum Stammordner Ihrer Anwendung im Amazon-S3-Bucket. Der Ordner sowie jegliche Unterordner sollten die ausführbare Build-Datei und alle unterstützenden Dateien enthalten.

Ein gültiger URI ist das Bucket-Präfix, das alle Dateien enthält, die zum Ausführen und Streamen der Anwendung benötigt werden. Beispiel: Ein Bucket namens mygamebuild enthält drei vollständige Versionen der Spiel-Build-Dateien, jede in einem separaten Ordner. Sie möchten den Build im Ordner mygamebuild-EN101 streamen. In diesem Beispiel ist der URI

Sie können dieses Feld nach dem Abschluss der Erstellung nicht mehr bearbeiten.

- **executable-path:**

Dies ist der relative Pfad und der Dateiname der ausführbaren Datei, die Amazon GameLift Streams streamen wird. Geben Sie einen Pfad relativ zu `application-source-uri`. Die Datei muss sich im Stammordner der Anwendung befinden. Für Windows-Anwendungen muss es sich bei der Datei um eine gültige ausführbare Windows-Datei oder Batchdatei handeln, deren Dateiname auf `.exe`, `.cmd` oder `.bat` endet. Für Linux-Anwendungen muss es sich bei der Datei um eine gültige ausführbare Linux-Binärdatei oder um ein Skript handeln, das eine erste Interpreterzeile enthält, die mit einem Shebang (`"`) beginnt. `#!`

Sie können dieses Feld nach dem Abschluss der Erstellung nicht mehr bearbeiten.

Wenn die Anfrage erfolgreich ist, gibt Amazon GameLift Streams eine Antwort zurück, die der folgenden ähnelt:

```
{  
    "Arn": "arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6",  
    "Description": "MyGame v1",  
    "RuntimeEnvironment": {  
        "Type": "PROTON",  
        "Version": "20241007"  
    },  
    "ExecutablePath": "launcher.exe",  
    "ApplicationSourceUri": "s3://amzn-s3-demo-bucket/example",  
    "Id": "a-9ZY8X7Wv6",  
    "Status": "PROCESSING",  
    "CreatedAt": "2022-11-18T15:47:11.924000-08:00",  
    "LastUpdatedAt": "2022-11-18T15:47:11.924000-08:00"  
}
```

Um den Status Ihrer Anwendung zu überprüfen, rufen Sie den [GetApplication](#)Befehl auf, wie im folgenden Beispiel gezeigt.

```
aws gameliftstreams get-application /  
--identifier a-9ZY8X7Wv6
```

Amazon GameLift Streams benötigt einige Minuten, um Ihre Bewerbung vorzubereiten. Während dieser Zeit befindet sich der neue Antrag im Status Bearbeitung. Wenn sich Ihre Bewerbung im Status Bereit befindet, können Sie mit dem nächsten Schritt fortfahren [Erstellen Sie eine Stream-Gruppe.](#)

Wenn die Anfrage einen Fehler zurückgibt oder wenn die Anwendung erstellt wurde, aber den Status Fehler hat, stellen Sie sicher, dass Sie mit Benutzeranmeldedaten arbeiten, die Zugriff auf Amazon S3 und Amazon GameLift Streams beinhalten.

Note

Wenn sich eine Anwendung im Status Bereit befindet, hat Amazon GameLift Streams Ihre Anwendungsdateien erfolgreich in seinen privaten Amazon S3 S3-Bucket kopiert. Sie können Ihre ursprünglichen Anwendungsdateien löschen, ohne dass sich dies auf Ihre neue Anwendung auswirkt. Dies hilft Ihnen auch bei der Optimierung der Speicherkosten. Weitere Informationen finden Sie unter [Löschen einer Anwendung](#).

Bearbeiten Sie eine Anwendung

Sie können die Einstellungen für jede Anwendung aktualisieren, die sich im Status Bereit befindet. Wenn Sie Änderungen an einer vorhandenen Anwendung vornehmen, wirken sich diese Änderungen auf das Streaming-Verhalten sowohl für neue als auch für bestehende Stream-Gruppen aus.

Console

Um eine Anwendung in der Amazon GameLift Streams-Konsole zu bearbeiten

1. Wählen Sie in der Navigationsleiste Anwendungen aus, um eine Liste Ihrer vorhandenen Anwendungen anzuzeigen. Wählen Sie die Anwendung aus, die Sie bearbeiten möchten.
2. Suchen Sie auf der Seite mit den Anwendungsdetails den Abschnitt, der die Einstellungen enthält, die Sie ändern möchten, und wählen Sie entsprechend „Tags bearbeiten“ oder „Tags verwalten“.
3. Sie können die folgenden Einstellungen ändern:

Short description

Dies ist eine von Menschen lesbare Bezeichnung für Ihre Anwendung. Dieser Wert muss nicht eindeutig sein. Es wird empfohlen, eine aussagekräftige Beschreibung, einen

aussagekräftigen Namen oder eine aussagekräftige Bezeichnung für die Anwendung zu verwenden. Sie können dieses Feld jederzeit bearbeiten.

Anwendungsprotokollpfad

Dies ist/sind der/die Pfad(e) zum Anwendungsordner oder zur Anwendungsdatei, der/die Protokolle enthält, die Sie speichern möchten. Geben Sie jeden Protokollpfad relativ zu Ihrem Anwendungsbasispfad an. Wenn Sie diese Funktion verwenden, kopiert Amazon GameLift Streams am Ende jeder Stream-Sitzung die von Ihnen angegebene (n) Datei (en) in den Amazon S3 S3-Bucket, den Sie benennen. Der Kopiervorgang wird nicht rekursiv in den Unterordnern eines Anwendungsordners ausgeführt.

Um die Protokollierung zu deaktivieren, entfernen Sie alle Anwendungsprotokollpfade und löschen Sie das Ausgabeziel für das Anwendungsprotokoll.

Sie können dieses Feld jederzeit bearbeiten.

Ausgabe des Anwendungsprotokolls

Dies ist der URI zum Amazon S3 S3-Bucket, in den Amazon GameLift Streams Anwendungsprotokolldateien kopiert. Dieses Feld ist erforderlich, wenn Sie einen Anwendungsprotokollpfad angeben.

Um die Protokollierung zu deaktivieren, entfernen Sie alle Anwendungsprotokollpfade und löschen Sie das Ausgabeziel für das Anwendungsprotokoll.

Sie können dieses Feld jederzeit bearbeiten.

Um Protokolldateien in Ihrem Namen zu speichern, muss Amazon GameLift Streams die Erlaubnis zum Speichern Ihres S3-Buckets erhalten. Wenn Sie Amazon GameLift Streams den Bucket für die Protokollierung erstellen lassen, wird die Berechtigungsrichtlinie bei der Erstellung automatisch angewendet. Wenn Sie Ihren eigenen Bucket bereitstellen, müssen Sie die Berechtigungsrichtlinie selbst anwenden.

Weitere Informationen finden Sie unter [Berechtigungsrichtlinie für das Anwendungsprotokoll-Bucket](#).

Tags (Markierungen)

Tags sind Bezeichnungen, die Ihnen bei der Organisation Ihrer AWS Ressourcen helfen können. Weitere Informationen finden Sie unter [Taggen Ihrer AWS-Ressourcen](#).

Um beispielsweise Anwendungsversionen nachzuverfolgen, verwenden Sie ein Tag wie `application-version : my-game-1121`.

4. Wählen Sie Änderungen speichern aus. Die Amazon GameLift Streams-Konsole kehrt zur Seite mit den Anwendungsdetails zurück und zeigt die aktualisierten Einstellungen an.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren AWS-Region. Anweisungen zur Einrichtung finden Sie unter [Laden Sie das herunter AWS CLI](#).

Um eine Anwendung mit dem zu bearbeiten AWS CLI

AWS CLI Verwenden Sie dabei den [UpdateApplication](#)Befehl, der an Ihren Inhalt angepasst ist.

```
aws gameliftstreams update-application \
--identifier a-9ZY8X7Wv6 \
--description "MyGame v2" \
--application-log-paths '[".\\logs"]' \
--application-log-output-uri "s3://amzn-s3-demo-bucket/mygame"
```

where

- **identifier:** Die zu bearbeitende Anwendung.

Dieser Wert ist ein [Amazon-Ressourcenname \(ARN\)](#) oder eine Amazon-ID, die die Anwendungsressource eindeutig identifiziert.

ARN-Beispiel: `arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6`

ID-Beispiel: `a-9ZY8X7Wv6`

- **description:**

Dies ist eine von Menschen lesbare Bezeichnung für Ihre Anwendung. Dieser Wert muss nicht eindeutig sein. Es wird empfohlen, eine aussagekräftige Beschreibung, einen aussagekräftigen

Namen oder eine aussagekräftige Bezeichnung für die Anwendung zu verwenden. Sie können dieses Feld jederzeit bearbeiten.

- **application-log-paths:**

Dies ist/sind der/die Pfad(e) zum Anwendungsordner oder zur Anwendungsdatei, der/die Protokolle enthält, die Sie speichern möchten. Geben Sie jeden Protokollpfad relativ zu Ihrem Anwendungsbasispfad an. Wenn Sie diese Funktion verwenden, kopiert Amazon GameLift Streams am Ende jeder Stream-Sitzung die von Ihnen angegebene (n) Datei (en) in den Amazon S3 S3-Bucket, den Sie benennen. Der Kopiervorgang wird nicht rekursiv in den Unterordnern eines Anwendungsordners ausgeführt.

Um die Protokollierung zu deaktivieren, entfernen Sie alle Anwendungsprotokollpfade und löschen Sie das Ausgabeziel für das Anwendungsprotokoll.

Sie können dieses Feld jederzeit bearbeiten.

- **application-log-output-uri:**

Dies ist der URI zum Amazon S3 S3-Bucket, in den Amazon GameLift Streams Anwendungsprotokolldateien kopiert. Dieses Feld ist erforderlich, wenn Sie einen Anwendungsprotokollpfad angeben.

Um die Protokollierung zu deaktivieren, entfernen Sie alle Anwendungsprotokollpfade und löschen Sie das Ausgabeziel für das Anwendungsprotokoll.

Sie können dieses Feld jederzeit bearbeiten.

Um Protokolldateien in Ihrem Namen zu speichern, muss Amazon GameLift Streams die Erlaubnis zum Speichern Ihres S3-Buckets erhalten. Wenn Sie Amazon GameLift Streams den Bucket für die Protokollierung erstellen lassen, wird die Berechtigungsrichtlinie bei der Erstellung automatisch angewendet. Wenn Sie Ihren eigenen Bucket bereitstellen, müssen Sie die Berechtigungsrichtlinie selbst anwenden.

Weitere Informationen finden Sie unter [Berechtigungsrichtlinie für das Anwendungsprotokoll-Bucket](#).

Löschen einer Anwendung

Löschen Sie eine Anwendung, wenn Sie sie nicht mehr benötigen. Diese Aktion löscht die Anwendung dauerhaft, einschließlich der in Amazon GameLift Streams gespeicherten

Anwendungsinhaltsdateien. Dadurch werden jedoch nicht die Originaldateien gelöscht, die Sie in Ihren Amazon S3 S3-Bucket hochgeladen haben. Sie können diese jederzeit löschen, nachdem Amazon GameLift Streams eine Anwendung erstellt hat. Dies ist das einzige Mal, dass Amazon GameLift Streams auf Ihren Amazon S3 S3-Bucket zugreift.

Sie können nur eine Anwendung löschen, die die folgenden Bedingungen erfüllt:

- Die Anwendung befindet sich im Status Bereit oder Fehler.
- Eine Anwendung streamt in keiner laufenden Stream-Sitzung. Sie müssen warten, bis der Client die Stream-Sitzung beendet, oder die Amazon GameLift Streams-API aufrufen [TerminateStreamSession](#), um den Stream zu beenden.

Wenn die Anwendung mit Stream-Gruppen verknüpft ist, müssen Sie sie von allen zugehörigen Stream-Gruppen trennen, bevor Sie sie löschen können. In der Konsole werden Sie in einem Dialogfeld durch diesen Vorgang geführt.

Console

Um eine Anwendung mithilfe der Amazon GameLift Streams-Konsole zu löschen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Wählen Sie in der Navigationsleiste Anwendungen aus, um eine Liste Ihrer vorhandenen Anwendungen anzuzeigen. Wählen Sie die Anwendung aus, die Sie löschen möchten.
3. Wählen Sie auf der Detailseite der Anwendung die Option Löschen aus.
4. Bestätigen Sie im Dialogfeld Löschen die Löschaktion.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren AWS-Region. Anweisungen zur Einrichtung finden Sie unter[Laden Sie das herunter AWS CLI](#).

Um eine Anwendung mit dem zu löschen AWS CLI

AWS CLI Verwenden Sie dabei den [DeleteApplication](#)Befehl, der an Ihren Inhalt angepasst ist.

```
aws gameliftstreams delete-application \
--identifier arn:aws:gameliftstreams:us-west-2:111122223333:application/
a-9ZY8X7Wv6
```

where

- **identifier:** Die zu löschen Anwendung.

Dieser Wert ist ein [Amazon-Ressourcenname \(ARN\)](#) oder eine Amazon-ID, die die Anwendungsressource eindeutig identifiziert.

ARN-Beispiel: arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6

ID-Beispiel: a-9ZY8X7Wv6

Amazon GameLift Streams beginnt mit dem Löschen der Anwendung. Während dieser Zeit befindet sich die Anwendung im Deleting Status. Nachdem Amazon GameLift Streams die Anwendung gelöscht hat, können Sie sie nicht mehr abrufen.

Berechtigungsrichtlinie für das Anwendungsprotokoll-Bucket

Wenn Sie Ihren eigenen Amazon S3 S3-Bucket mit Anwendungsprotokoll bereitstellen, müssen Sie eine Berechtigungsrichtlinie auf den Bucket anwenden, damit Amazon GameLift Streams Protokolldateien im Bucket speichern kann. Verwenden Sie die folgende Vorlage, um die Berechtigungen in Amazon S3 zu aktualisieren.

Vorlage für eine Bucket-Berechtigungsrichtlinie

Kopieren Sie den folgenden Richtliniencode und wenden Sie ihn auf den Bucket an, den Sie für die Anwendungsprotokolle verwenden möchten. Achten Sie darauf, amzn-s3-demo-bucket durch den Namen des vorhandenen S3-Buckets zu ersetzen.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PutPolicy",
      "Effect": "Allow",
      "Principal": {
```

```
        "Service": [
            "gameliftstreams.amazonaws.com"
        ],
    },
    "Action": "s3:PutObject",
    "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "your 12-digit account id"
        }
    }
}
]
```

Note

Amazon GameLift Streams erlaubt keinen kontoübergreifenden Zugriff auf Ressourcen. Der Amazon S3 S3-Bucket muss demselben AWS Konto gehören wie die Anwendungsressource. Obwohl dies vom Service strikt durchgesetzt wird, ist es eine bewährte Methode, bei der Erteilung von Genehmigungen für einen AWS Dienst immer `aws:SourceArn` Bedingungen mit `aws:SourceAccount` einzubeziehen, um [das Problem des verwirrten Stellvertreters](#) zu vermeiden.

Verknüpfte Stream-Gruppen

Wenn Sie mehrere Anwendungen streamen möchten, indem Sie denselben Pool von Rechenressourcen verwenden, können Sie mehrere Anwendungen mit derselben Stream-Gruppe verknüpfen. Ebenso können Sie eine Anwendung mit mehreren Stream-Gruppen verknüpfen, wenn Sie eine Anwendung streamen möchten, indem Sie verschiedene Gruppen von Rechenressourcen verwenden.

Weitere Informationen zum Verknüpfen von Anwendungen mit Stream-Gruppen finden Sie unter [Übersicht über Stream-Gruppen für mehrere Anwendungen](#).

Streaming mit einer Amazon GameLift Streams-Stream-Gruppe verwalten

Nachdem Sie eine Amazon GameLift Streams-Anwendung eingerichtet haben, können Sie Rechenressourcen verwalten und bereitstellen, um Ihre Anwendung auszuführen und zu streamen. Eine Amazon GameLift Streams-Stream-Gruppe stellt eine Sammlung dieser Rechenressourcen dar. Sie geben die maximale Anzahl gleichzeitiger Streams an, die unterstützt werden sollen, indem Sie die Stream-Kapazität skalieren.

Amazon GameLift Streams weist Rechenressourcen dort zu, AWS-Region wo Sie eine Stream-Gruppe erstellen. Sie können einer Stream-Gruppe auch Remote-Standorte hinzufügen und die Kapazität pro Standort verwalten. Es hat sich bewährt, Stream-Sitzungen an Orten abzuhalten, die sich geografisch in der Nähe Ihrer Endbenutzer befinden. Dies trägt dazu bei, die Latenz zu minimieren und die Streamqualität zu verbessern. Weitere Informationen finden Sie unter [AWS-Regionen und entfernte Standorte, die von Amazon GameLift Streams unterstützt werden](#).

In einer Stream-Gruppe können Sie eine oder mehrere Amazon GameLift Streams-Anwendungen angeben, die die Stream-Gruppe streamen kann. Eine einzelne Anwendung kann sich in mehreren Stream-Gruppen befinden, sodass Sie unterschiedliche Konfigurationen oder Typen von Rechenressourcen einrichten können, um dieselbe Anwendung zu streamen. Um beispielsweise zwei Optionen für die Grafikqualität für das Streaming einer Anwendung bereitzustellen, können Sie zwei Stream-Gruppen mit unterschiedlichen Stream-Klassenkonfigurationen einrichten und sie mit derselben Anwendung verknüpfen.

Umgekehrt kann eine einzelne Stream-Gruppe mehrere Anwendungen haben: die Standardanwendung, die Sie bei der Erstellung der Stream-Gruppe festlegen können, und weitere verknüpfte Anwendungen. Weitere Informationen finden Sie unter [Übersicht über Stream-Gruppen für mehrere Anwendungen](#).

Wie Sie Ihre Stream-Gruppen und Anwendungen miteinander in Beziehung setzen, hängt von Ihrem Anwendungsfall ab, aber die Beziehung kann sein many-to-many.

Stream-Gruppen sollten alle 3-4 Wochen neu erstellt werden, um wichtige Service-Updates und Fixes zu erhalten. Weitere Informationen finden Sie unter [Lebenszyklus einer Stream-Gruppe](#).

Themen

- [Über die Stream-Kapazität](#)
- [Kapazität und Servicekontingenten](#)

- [Über Standorte](#)
- [Erstellen Sie eine Stream-Gruppe](#)
- [Bearbeiten Sie die allgemeinen Einstellungen](#)
- [Kapazität bearbeiten](#)
- [Verhalten beim Herunterskalieren der Kapazität](#)
- [Fügen Sie Standorte zu einer Stream-Gruppe hinzu](#)
- [Standorte in einer Stream-Gruppe entfernen](#)
- [Löschen Sie eine Stream-Gruppe](#)
- [Verknüpfte Anwendungen](#)
- [Lebenszyklus einer Stream-Gruppe](#)
- [Wartung der Stream-Gruppe](#)

Über die Stream-Kapazität

Sie verwalten die Anzahl der Streams, die Sie Endbenutzern gleichzeitig bereitstellen können, indem Sie die Kapazität oder Stream-Kapazität der Stream-Gruppe festlegen. Die Stream-Kapazität stellt die Anzahl gleichzeitiger Stream-Sitzungen dar, die eine Stream-Gruppe unterstützen kann. Sie wird an jedem Standort konfiguriert.

- Always-On-Kapazität: Wenn diese Einstellung ungleich Null ist, gibt sie die minimale Streaming-Kapazität an, die Ihnen zugewiesen und nie wieder an den Service zurückgegeben wird. Sie zahlen jederzeit für diese Basiskapazität, unabhängig davon, ob sie genutzt oder nicht genutzt wird.
- Maximale Kapazität: Dies gibt die maximale Kapazität an, die der Service Ihnen zuweisen kann. Der Start neu erstellter Streams kann einige Minuten in Anspruch nehmen. Im Leerlauf wird Kapazität wieder an den Service zurückgegeben. Sie zahlen für die Ihnen zugewiesene Kapazität, bis sie freigegeben wird.
- Zielkapazität im Leerlauf: Dies gibt ungenutzte Kapazität an, die der Service Ihnen im Voraus zuweist und für Sie im Vorgriff auf zukünftige Aktivitäten bereithält. Dies trägt dazu bei, Ihre Benutzer vor Verzögerungen bei der Kapazitätszuweisung zu schützen. Sie zahlen für Kapazität, die in diesem intendierten Ruhezustand gehalten wird.

Wenn Sie eine Stream-Gruppe mit einer maximalen Kapazität von 100 an einem Standort haben, bedeutet dies, dass die Stream-Gruppe über genügend Ressourcen verfügt, um an diesem Standort

100 Endbenutzer gleichzeitig zu streamen. Sie können die Stream-Kapazität jederzeit an jedem Standort erhöhen oder verringern (bis zu Ihrem aktuellen Kontingent), um den Änderungen der Benutzernachfrage gerecht zu werden.

Amazon GameLift Streams versucht zunächst, neue Sitzungsanfragen mit ungenutzter Kapazität zu erfüllen, die Ihnen bereits zugewiesen wurde. Wenn dies dazu führt, dass die Menge der ungenutzten Kapazität unter Ihre Zielkapazität im Leerlauf fällt, wird neue Kapazität in asynchroner Weise zugewiesen. Wenn keine ungenutzte Kapazität verfügbar ist, wird die Anforderung angehalten, während neue Kapazität on-demand zugewiesen wird, bis die maximale Kapazität für die Stream-Gruppe erreicht ist. Wenn das Maximum erreicht ist und immer noch keine ungenutzte Kapazität verfügbar ist, wartet die Sitzungsanforderung darauf, dass eine bestehende Sitzung beendet und Kapazität freigegeben wird.

Wenn Sitzungen beendet werden, wird die entsprechende Kapazität als ungenutzt markiert. Wenn mehr ungenutzte Kapazität vorhanden ist als der dafür angestrebte Wert vorgibt, wird die überschüssige Kapazität freigegeben und nach einer kurzen Verzögerung wieder an den Service zurückgegeben. Der Service gibt ungenutzte Kapazität nicht frei, wenn Ihre Kapazität dadurch unter das konfigurierte Minimum sinken würde (das gleich Null sein könnte).

Bei der Angabe der Stream-Kapazität in Stream-Gruppen mit Stream-Klassen mit mehreren Mandanten (die mehr als eine Sitzung pro Rechenressource streamen können), muss die Kapazität ein Vielfaches der Tenancy sein. Beispielsweise hat die gen6n_high Stream-Klasse eine Mehrmandantenfähigkeit von 2. Das bedeutet, dass jede Rechenressource, die Ihrer Stream-Gruppe zugewiesen wird, an 2 Clients streamen kann. Daher muss die von Ihnen angeforderte Kapazität ein Vielfaches von 2 sein.

Die Skalierung der Kapazität spiegelt sich in Ihren Gesamtkosten für die Stream-Gruppe wider. Stellen Sie sicher, dass Sie Abrechnungsbenachrichtigungen einrichten, um Ihre Amazon GameLift Streams-Kosten zu verwalten. Weitere Informationen finden Sie unter [Erstellen Sie Abrechnungsbenachrichtigungen, um die Nutzung zu überwachen](#).

Um die Kapazität der Stream-Gruppe zu ändern, bearbeiten Sie Ihre Stream-Gruppeneinstellungen und geben Sie neue Werte für die Kapazitätseinstellungen ein. Wenn Sie die Always-On-Kapazität ändern, passt Amazon GameLift Streams die zugewiesenen Ressourcen an den neuen Wert an, indem neue Ressourcen bereitgestellt oder bestehende heruntergefahren werden. Die Erhöhung der Always-On-Kapazität kann mehr als ein paar Minuten dauern, wenn Ressourcen nicht sofort verfügbar sind. Eine Verringerung der Always-On-Kapazität dauert einige Minuten, bis die Bereitstellung zugewiesener Ressourcen aufgehoben ist.

Beispiel: Stream-Kapazitätskonfigurationen

Die folgenden Beispiele zeigen gängige Stream-Kapazitätskonfigurationen für verschiedene Anwendungsfälle:

1. Kostenbewusste Entwicklungsphase: Sie sind ein Entwickler, der Kosten sparen möchte. Sie setzen Minimum (always-on) capacity = 0, Maximum capacity = 10 und Target Idle (pre-warmed) capacity = 1. Dadurch bleibt mindestens eine Sitzung für den Schnellstart verfügbar.
2. Geplante Veranstaltung mit fester Nachfrage: Sie möchten schnelle Sitzungsstarts für eine geplante Veranstaltung mit bekannter Nachfrage. Sie legen Minimum (always-on) capacity = 200, Maximum capacity = 200 und Target Idle (pre-warmed) capacity = 0 fest. Sie zahlen nur für 200 Kapazitäten. Es treten keine Verzögerungen bei der Skalierung auf, da der Bedarf bekannt ist.
3. Großveranstaltung mit hoher Kapazität: Sie planen 1.000 Benutzer mit 100 neuen Sitzungen pro Minute zu Spitzenzeiten. Sie legen Minimum = 0, Maximum = 1.000 und Target Idle = 100 fest. Das spart Geld im Leerlauf. Dadurch stehen mindestens 100 Sitzungen für einen schnellen Start zur Verfügung.

 Note

Der OnDemandCapacity Eingabeparameter ist veraltet. Verwenden Sie ihn MaximumCapacity stattdessen, wenn Sie die Kapazität über die API konfigurieren.

Kapazität und Servicekontingenten

Die Nutzung von Amazon GameLift Streams unterliegt Servicekontingenten, die die Gesamtzahl der GPUs (Rechenressourcen) begrenzen, die Sie für das Streaming in Ihrem Konto konfigurieren können. Die Standardkontingente und die Nutzung der Kontingente können in der Service Quota Console für GameLift Streams eingesehen werden. Wenn Sie wissen, wie diese Kontingente mit der Streamingkapazität interagieren, können Sie Ihre Streaming-Infrastruktur besser planen und Kapazitätsbeschränkungen vermeiden.

Genauer gesagt geben die GPU-Dienstkontingente die maximale GPUs Anzahl einer bestimmten Stream-Klassenfamilie an, die Sie pro Standort für alle Stream-Gruppen in Ihrem Konto anfordern können. Wenn Ihr Konto beispielsweise ein Limit von 5 gen6n GPUs Zoll hatus-west-2, muss die

Summe der Daten, die zur Bereitstellung der gesamten Stream-Kapazität us-west-2 für alle Ihre Stream-Gruppen gen6n GPUs erforderlich sind, kleiner oder gleich 5 sein. Dies gilt sowohl GPUs für ständig verfügbare als auch für On-Demand-Kapazität.

Bei der Berechnung der gesamten Stream-Kapazität, die von diesen bereitgestellt wird GPUs, ist zu beachten, dass Stream-Klassen mit mehreren Mandanten das Streaming von mehr als einer Sitzung pro GPU unterstützen. Wenn Sie beispielsweise in Ihren Stream-Gruppen Stream-Klassen mit mehreren Mandanten verwenden, müssen Sie dies berücksichtigen gen6n_high, wenn Sie festlegen, wie die Kapazität auf Ihr Kontingent angerechnet wird. Single-Tenant-Stream-Klassen, wie z. B. gen6n_ultra und gen6n_ultra_win2022, weisen pro Stream-Sitzung eine GPU zu.

Beispiel: Wie sich Kontingente auf die Kapazität auswirken

Das folgende Beispiel zeigt, wie Servicekontingente mit der Stream-Kapazität mehrerer Streamgruppen und Standorte interagieren. Gehen Sie in diesem Beispiel davon aus, dass Ihr Konto eine Quote von 10 gen6n GPUs pro Standort hat.

1. Erstellen Sie eine Streamgruppe für einen Mandanten: Sie erstellen eine Streamgruppe mithilfe der gen6n_ultra Stream-Klasse mit einer Gesamtkapazität von 5 (ständig verfügbar plus On-Demand). us-east-2 Da diese Stream-Klasse über eine 1:1 -Tenancy (1 Stream pro GPU) verfügt, benötigen Sie 5 GPUs für eine Gesamtkapazität von 5. Somit bleiben Ihnen noch 5 übrig GPUs . us-east-2
2. Erstellen Sie eine Streamgruppe mit mehreren Mandanten: Sie erstellen eine weitere Streamgruppe mithilfe der gen6n_high Stream-Klasse mit einer Gesamtkapazität von 6 us-east-2 Da diese Stream-Klasse über eine 1:2 -Tenancy (2 Streams pro GPU) verfügt, benötigen Sie nur 3 GPUs für eine Gesamtkapazität von 6. Somit bleiben Ihnen noch 2 übrig GPUs . us-east-2
3. Kapazität an anderen Speicherorten hinzufügen: Nachdem Sie diese Stream-Gruppen erstellt haben, sind noch 2 GPUs verfügbar us-east-2, aber Sie haben immer noch 10 an anderen Speicherorten GPUs verfügbar, z. B. us-west-2 oder eu-west-1. Sie können diese Standorte zu einer der Streamgruppen hinzufügen, die Sie zuvor erstellt haben, oder neue Stream-Gruppen mit diesen Speicherorten erstellen.

Dieses Beispiel zeigt, dass Kontingente pro Standort und für alle Ihre Stream-Gruppen gelten, sodass Sie Ihre Streaming-Kapazität auf mehrere geografische Regionen verteilen und gleichzeitig die Dienstgrenzen einhalten können.

Note

Sie können Ihre angewendete Kontoebene oder Ihr Standardkontingent, einschließlich der Nutzung dieser Kontingente, in der Konsole Service Quotas anzeigen, indem Sie GameLift Streams als AWS Service auswählen. Weitere Informationen finden Sie unter [Amazon GameLift Streams-Servicekontingente](#).

Über Standorte

An diesem Ort weist Amazon GameLift Streams Rechenressourcen zu, um Ihre Anwendung zu hosten und Benutzern zu streamen. Für eine geringere Latenz und eine bessere Qualität sollten Sie Standorte wählen, die sich näher an Ihren Benutzern befinden. Standardmäßig können Sie von dem Ort aus streamen, AWS-Region an dem Sie Ihre Stream-Gruppe erstellt haben, der als primärer Standort bezeichnet wird. Darüber hinaus kann eine Stream-Gruppe ihre Reichweite auf Streams von anderen unterstützten Standorten, sogenannten Remote-Standorten, ausdehnen.

Eine vollständige Liste der unterstützten Standorte finden Sie unter [AWS-Regionen und abgelegene Standorte](#).

Stream-Gruppe mit mehreren Standorten

Eine Stream-Gruppe, die so konfiguriert ist, dass sie neben dem primären Standort (dem, an AWS-Region dem Sie die Streamgruppe erstellt haben) auch Anwendungen und Stream-Sitzungen von mehreren Standorten aus hostet. Sie verwalten die Kapazität für jeden Standort.

Erstellen Sie eine Stream-Gruppe

Console

Um eine Stream-Gruppe in der Amazon GameLift Streams-Konsole zu erstellen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#). Wählen Sie den AWS-Region Ort aus, an dem Sie Ihre Stream-Gruppe erstellen möchten. Diese Region muss mit der Region der Anwendung identisch sein, die Sie mit der Stream-Gruppe streamen möchten. Weitere Informationen finden Sie im Handbuch AWS-Managementkonsole Erste Schritte unter [Auswahl einer Region](#).

2. Um den Workflow zur Erstellung zu öffnen, wählen Sie im Navigationsbereich Stream-Gruppen und dann Stream-Gruppe erstellen aus.
3. Geben Sie unter Stream-Gruppe definieren Folgendes ein:
 - a. Beschreibung

Dies ist eine von Menschen lesbare Bezeichnung für Ihre Stream-Gruppe. Dieser Wert muss nicht eindeutig sein. Es wird empfohlen, eine aussagekräftige Beschreibung, einen aussagekräftigen Namen oder eine aussagekräftige Bezeichnung für die Stream-Gruppe zu verwenden. Sie können dieses Feld jederzeit bearbeiten.

- b. Tags
- Tags sind Beschriftungen, die Ihnen bei der Organisation Ihrer AWS Ressourcen helfen können. Weitere Informationen finden Sie unter [Tagging von AWS-Ressourcen](#).
4. Wählen Sie unter Stream-Klasse auswählen eine Stream-Klasse für die Stream-Gruppe aus.
 - Optionen für die Stream-Klasse

Die Art der Rechenressourcen, mit denen Anwendungen ausgeführt und gestreamt werden sollen. Die Stream-Klasse wirkt sich auf die Qualität des Streaming-Vorgangs und auf die Kosten aus. Sie können pro Stream-Gruppe nur eine Stream-Klasse angeben. Wählen Sie die Klasse, die am besten zu Ihrer Anwendung passt.

Stream-Klasse	Description
gen6n_pro _win2022	(NVIDIA, pro) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität, die maximale Ressourcen benötigen. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12. Kompatibel mit Unreal Engine-Versionen bis 5.6, 32- und 64-Bit-Anwendungen und Anti-Cheat-Technologie. Verwendet die NVIDIA L4 Tensor Core-GPU. Ressourcen pro Anwendung: vCPUs: 16. RAM: 64 GB. VRAM: 24 GB. Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.

Stream-Klasse	Description
gen6n_pro	<p>(NVIDIA, pro) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität, die maximale Ressourcen benötigen. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 16. RAM: 64 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>
gen6n_ult ra_win2022	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12. Kompatibel mit Unreal Engine-Versionen bis 5.6, 32- und 64-Bit-Anwendungen und Anti-Cheat-Technologie. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>
gen6n_ultra	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>

Stream-Klasse	Description
gen6n_high	<p>(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer moderate-to-high 3D-Szene. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 4. RAM: 16 GB. VRAM: 12 GB.</p> <p>Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.</p>
gen6n_medium	<p>(NVIDIA, medium) Unterstützt Anwendungen mit mittlerer bis hoher 3D-Szenenkomplexität. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 2. RAM: 8 GB. VRAM: 6 GB.</p> <p>Tenancy: Unterstützt bis zu vier gleichzeitige Stream-Sitzungen.</p>
gen6n_small	<p>(NVIDIA, small) Unterstützt Anwendungen mit geringer 3D-Szenenkomplexität und geringer CPU-Auslastung. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 1. RAM: 4 GB. VRAM: 2 GB.</p> <p>Tenancy: Unterstützt bis zu zwölf gleichzeitige Stream-Sitzungen.</p>

Stream-Klasse	Description
gen5n_win2022	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12 und DirectX 11. Unterstützt Unreal Engine bis Version 5.6, 32- und 64-Bit-Anwendungen und Anti-Cheat-Technologie. Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>
gen5n_high	<p>(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer moderate-to-high 3D-Szene. Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 4. RAM: 16 GB. VRAM: 12 GB.</p> <p>Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.</p>
gen5n_ultra	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität. Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>

Stream-Klasse	Description
gen4n_win2022	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12 und DirectX 11. Unterstützt Unreal Engine bis Version 5.6, 32- und 64-Bit-Anwendungen und Anti-Cheat-Technologie. Verwendet die NVIDIA T4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 16 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>
gen4n_high	<p>(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer moderate-to-high 3D-Szene. Verwendet die NVIDIA T4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 4. RAM: 16 GB. VRAM: 8 GB.</p> <p>Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.</p>
gen4n_ultra	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Verwendet die NVIDIA T4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 16 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>

Wählen Sie Next, um fortzufahren.

5. Wählen Sie unter Anwendung verknüpfen eine Anwendung aus, die Sie streamen möchten, oder wählen Sie "Keine Anwendung", um zu einem späteren Zeitpunkt eine Anwendung auszuwählen. Sie können die Stream-Gruppe bearbeiten, nachdem sie erstellt wurde, um

Anwendungen hinzuzufügen oder zu entfernen. Sie können nur eine Anwendung verknüpfen, die sich im Ready Status befindet und deren Laufzeit mit der von Ihnen ausgewählten Stream-Klasse kompatibel ist. Standardmäßig sind dies die einzigen Anwendungen, die in der Tabelle aufgeführt sind. Um alle Anwendungen im Ready Status zu sehen, wählen Sie All runtimes in der Drop-down-Liste aus.

 Note

Wenn Ihre Bewerbung nicht aufgeführt ist, überprüfen Sie die aktuelle AWS-Region Einstellung. Sie können eine Anwendung nur mit einer Stream-Gruppe verknüpfen, die sich in derselben Region befindet.

Wählen Sie Next, um fortzufahren.

6. Wählen Sie unter Stream-Einstellungen konfigurieren unter Standorte und Kapazität einen oder mehrere Standorte aus, an denen Ihre Stream-Gruppe die Kapazität zum Streamen Ihrer Anwendung haben soll. Standardmäßig wurde die Region, in der Sie die Stream-Gruppe erstellen, der so genannte primäre Standort, bereits zu Ihrer Stream-Gruppe hinzugefügt und kann nicht entfernt werden. Du kannst weitere Standorte hinzufügen, indem du das Kästchen neben jedem Standort markierst, den du hinzufügen möchtest. Für eine geringere Latenz und eine bessere Streaming-Qualität sollten Sie Standorte wählen, die sich in der Nähe Ihrer Benutzer befinden.

Für jeden Standort können Sie seine Streaming-Kapazität angeben. Die Stream-Kapazität ist die Anzahl der gleichzeitigen Streams, die zu einem bestimmten Zeitpunkt aktiv sein können. Sie legen die Stream-Kapazität pro Speicherort in jeder Stream-Gruppe fest.

- Always-On-Kapazität: Wenn diese Einstellung ungleich Null ist, gibt sie die minimale Streaming-Kapazität an, die Ihnen zugewiesen und nie wieder an den Service zurückgegeben wird. Sie zahlen jederzeit für diese Basiskapazität, unabhängig davon, ob sie genutzt oder nicht genutzt wird.
- Maximale Kapazität: Dies gibt die maximale Kapazität an, die der Service Ihnen zuweisen kann. Der Start neu erstellter Streams kann einige Minuten in Anspruch nehmen. Im Leerlauf wird Kapazität wieder an den Service zurückgegeben. Sie zahlen für die Ihnen zugewiesene Kapazität, bis sie freigegeben wird.
- Zielkapazität im Leerlauf: Dies gibt ungenutzte Kapazität an, die der Service Ihnen im Voraus zuweist und für Sie im Vorgriff auf zukünftige Aktivitäten bereithält. Dies trägt dazu

bei, Ihre Benutzer vor Verzögerungen bei der Kapazitätszuweisung zu schützen. Sie zahlen für Kapazität, die in diesem intendierten Ruhezustand gehalten wird.

Sie können Ihre gesamte Stream-Kapazität jederzeit erhöhen oder verringern, um der veränderten Nutzernachfrage nach einem Standort gerecht zu werden, indem Sie eine der Kapazitäten anpassen. Amazon GameLift Streams erfüllt Streaming-Anfragen mithilfe der inaktiven, vorab zugewiesenen Ressourcen im Always-On-Kapazitätspool, sofern welche verfügbar sind. Wenn die gesamte Always-On-Kapazität genutzt wird, stellt Amazon GameLift Streams zusätzliche Rechenressourcen bis zu der in On-Demand-Kapazität angegebenen Höchstzahl bereit. Wenn die zugewiesene Kapazität skaliert wird, spiegelt sich die Änderung in Ihren Gesamtkosten für die Stream-Gruppe wider.

Verknüpfte Anwendungen werden automatisch an jeden aktivierten Standort repliziert. Eine Anwendung muss die Replikation an einem Remote-Standort abgeschlossen haben, bevor der Remote-Standort einen Stream hosten kann. Um den Replikationsstatus zu überprüfen, öffnen Sie die Stream-Gruppe, nachdem sie erstellt wurde, und schlagen Sie in der Tabelle der verknüpften Anwendungen in der Spalte Replikationsstatus nach. Klicken Sie auf den aktuellen Status, um den Replikationsstatus für jeden hinzugefügten Standort zu sehen.

 Note

Anwendungsdaten werden an allen aktivierten Speicherorten gespeichert, einschließlich des primären Speicherorts für diese Stream-Gruppe. Stream-Sitzungsdaten werden sowohl am primären Standort als auch an dem Ort gespeichert, an dem das Streaming stattgefunden hat.

7. Überprüfen Sie unter Stream-Gruppe überprüfen und erstellen die Konfiguration Ihrer Stream-Gruppe und nehmen Sie bei Bedarf Änderungen vor. Wenn alles korrekt ist, wähle Stream-Gruppe erstellen.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren AWS-Region. Anweisungen zur Einrichtung finden Sie unter [Laden Sie das herunter AWS CLI](#).

Um eine Stream-Gruppe mit dem zu erstellen AWS CLI

AWS CLI Verwenden Sie dabei den [CreateStreamGroup](#)Befehl, der an Ihren Inhalt angepasst ist.

```
aws gameliftstreams create-stream-group \
--description "Test_gen4_high" \
--default-application-identifier arn:aws:gameliftstreams:us-
west-2:111122223333:application/a-9ZY8X7Wv6 \
--stream-class gen4n_high \
--location-configurations '[{"LocationName": "us-east-1", "AlwaysOnCapacity": 2,
"MaximumCapacity": 6, "TargetIdleCapacity": 1}]'
```

where

description:

Dies ist eine von Menschen lesbare Bezeichnung für Ihre Stream-Gruppe. Dieser Wert muss nicht eindeutig sein. Es wird empfohlen, eine aussagekräftige Beschreibung, einen aussagekräftigen Namen oder eine aussagekräftige Bezeichnung für die Stream-Gruppe zu verwenden. Sie können dieses Feld jederzeit bearbeiten.

default-application-identifier

Der Wert oder die ID des [Amazon-Ressourcennamens \(ARN\)](#), die einer Amazon GameLift Streams-Anwendungsressource zugewiesen wurde. Die Anwendung muss den READY Status haben.

ARN-Beispiel: arn:aws:gameliftstreams:us-west-2:111122223333:application/
a-9ZY8X7Wv6

ID-Beispiel: a-9ZY8X7Wv6

stream-class

Optionen für die Stream-Klasse

Die Art der Rechenressourcen, mit denen Anwendungen ausgeführt und gestreamt werden sollen. Die Stream-Klasse wirkt sich auf die Qualität des Streaming-Vorgangs und auf die Kosten aus. Sie können pro Stream-Gruppe nur eine Stream-Klasse angeben. Wählen Sie die Klasse, die am besten zu Ihrer Anwendung passt.

Stream-Klasse	Description
gen6n_pro _win2022	<p>(NVIDIA, pro) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität, die maximale Ressourcen benötigen. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12. Kompatibel mit Unreal Engine-Versionen bis 5.6, 32- und 64-Bit-Anwendungen und Anti-Cheat-Technologie. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 16. RAM: 64 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>
gen6n_pro	<p>(NVIDIA, pro) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität, die maximale Ressourcen benötigen. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 16. RAM: 64 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>
gen6n_ult ra_win2022	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12. Kompatibel mit Unreal Engine-Versionen bis 5.6, 32- und 64-Bit-Anwendungen und Anti-Cheat-Technologie. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>
gen6n_ultra	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Verwendet die NVIDIA L4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p>

Stream-Klasse	Description
	Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.
gen6n_high	(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer moderate-to-high 3D-Szene. Verwendet die NVIDIA L4 Tensor Core-GPU. Ressourcen pro Anwendung: vCPUs: 4. RAM: 16 GB. VRAM: 12 GB. Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.
gen6n_medium	(NVIDIA, medium) Unterstützt Anwendungen mit mittlerer bis hoher 3D-Szenenkomplexität. Verwendet die NVIDIA L4 Tensor Core-GPU. Ressourcen pro Anwendung: vCPUs: 2. RAM: 8 GB. VRAM: 6 GB. Tenancy: Unterstützt bis zu vier gleichzeitige Stream-Sitzungen.
gen6n_small1	(NVIDIA, small) Unterstützt Anwendungen mit geringer 3D-Szenenkomplexität und geringer CPU-Auslastung. Verwendet die NVIDIA L4 Tensor Core-GPU. Ressourcen pro Anwendung: vCPUs: 1. RAM: 4 GB. VRAM: 2 GB. Tenancy: Unterstützt bis zu zwölf gleichzeitige Stream-Sitzungen.

Stream-Klasse	Description
gen5n_win2022	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12 und DirectX 11. Unterstützt Unreal Engine bis Version 5.6, 32- und 64-Bit-Anwendungen und Anti-Cheat-Technologie. Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>
gen5n_ultra	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit extrem hoher 3D-Szenenkomplexität. Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 24 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>
gen5n_high	<p>(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer moderate-to-high 3D-Szene. Verwendet die NVIDIA A10G Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 4. RAM: 16 GB. VRAM: 12 GB.</p> <p>Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.</p>

Stream-Klasse	Description
gen4n_win2022	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Führt Anwendungen auf Microsoft Windows Server 2022 Base aus und unterstützt DirectX 12 und DirectX 11. Unterstützt Unreal Engine bis Version 5.6, 32- und 64-Bit-Anwendungen und Anti-Cheat-Technologie. Verwendet die NVIDIA T4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 16 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>
gen4n_ultra	<p>(NVIDIA, ultra) Unterstützt Anwendungen mit hoher 3D-Szenenkomplexität. Verwendet die NVIDIA T4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 8. RAM: 32 GB. VRAM: 16 GB.</p> <p>Tenancy: Unterstützt eine einzelne gleichzeitige Stream-Sitzung.</p>
gen4n_high	<p>(NVIDIA, hoch) Unterstützt Anwendungen mit komplexer moderate-to-high 3D-Szene. Verwendet die NVIDIA T4 Tensor Core-GPU.</p> <p>Ressourcen pro Anwendung: vCPUs: 4. RAM: 16 GB. VRAM: 8 GB.</p> <p>Tenancy: Unterstützt bis zu zwei gleichzeitige Stream-Sitzungen.</p>

location-configurations

Eine Reihe von Standorten, die dieser Stream-Gruppe hinzugefügt werden sollen, und deren Kapazitäten. Wenn keine Kapazitäten angegeben sind, weist Amazon GameLift Streams standardmäßig nur genügend Always-On-Stream-Kapazität zu, um einen Stream an dem Ort zu starten, an dem die Stream-Gruppe erstellt wurde. Eine vollständige Liste der Standorte, die Amazon GameLift Streams unterstützt, finden Sie unter [AWS-Regionen und entfernte Standorte, die von Amazon GameLift Streams unterstützt werden](#).

Die Kapazitätswerte müssen ein ganzzahliges Vielfaches des Tenancy-Werts der Stream-Klasse der Stream-Gruppe sein.

Wenn die Anfrage erfolgreich ist, gibt Amazon GameLift Streams eine Antwort zurück, die der folgenden ähnelt:

```
{  
    "Arn": "arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/  
    sg-1AB2C3De4",  
    "Description": "Test_gen4_high",  
    "DefaultApplication": {  
        "Id": "a-9ZY8X7Wv6"  
    },  
    "StreamClass": "gen4n_high",  
    "Id": "sg-1AB2C3De4",  
    "Status": "ACTIVATING",  
    "LastUpdatedAt": "2024-11-18T15:49:01.482000-08:00",  
    "CreatedAt": "2024-11-18T15:49:01.482000-08:00"  
}
```

Amazon GameLift Streams beginnt mit der Suche nach nicht zugewiesenen Rechenressourcen und deren Bereitstellung für die neue Stream-Gruppe, was mehrere Minuten dauern kann. Während dieser Zeit befindet sich die neue Stream-Gruppe im Status Aktiviert.

Sie können die Kapazität der Stream-Gruppe anpassen, wenn ihr Status Aktiv ist. Weitere Informationen finden Sie unter [Kapazität bearbeiten](#).

Wenn sich die Stream-Gruppe im Status Aktiv befindet, ist sie bereit, Ressourcen für das Streaming bereitzustellen. Informationen zum Starten des Streamings finden Sie unter [Stream-Sitzungen mit Amazon GameLift Streams starten](#).

Bearbeiten Sie die allgemeinen Einstellungen

Amazon GameLift Streams gruppiert die folgenden Einstellungen in der Konsole unter Stream-Gruppeneinstellungen: Status, Stream-Gruppen-ID, Beschreibung, Stream-Gruppen-ARN und Stream-Klasse. Von diesen ist Description die einzige, die Sie aktualisieren können, ohne eine neue Stream-Gruppe zu erstellen.

Console

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Wählen Sie in der Navigationsleiste Stream-Gruppen aus, um eine Liste Ihrer vorhandenen Stream-Gruppen anzuzeigen. Wählen Sie die Stream-Gruppe aus, die Sie bearbeiten möchten.
3. Wählen Sie auf der Detailseite der Stream-Gruppe die Option Einstellungen bearbeiten aus.
4. Um die Beschreibung zu aktualisieren, geben Sie einen neuen Wert ein.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren AWS-Region. Anweisungen zur Einrichtung finden Sie unter[Laden Sie das herunter AWS CLI](#).

Um die Beschreibung einer Stream-Gruppe zu bearbeiten, verwenden Sie den AWS CLI

AWS CLI Verwenden Sie dabei den [UpdateStreamGroup](#)Befehl, der an Ihren Inhalt angepasst ist.

```
aws gameliftstreams update-stream-group \
  --identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4 \
  --description "MyGame - Ultra"
```

where

identifier

Ein [Amazon-Ressourcenname \(ARN\)](#) oder eine Amazon-ID, die die Stream-Gruppenressource eindeutig identifiziert.

ARN-Beispiel: arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4

ID-Beispiel: sg-1AB2C3De4

description

Dies ist eine von Menschen lesbare Bezeichnung für Ihre Stream-Gruppe. Dieser Wert muss nicht eindeutig sein. Es wird empfohlen, eine aussagekräftige Beschreibung, einen aussagekräftigen Namen oder eine aussagekräftige Bezeichnung für die Stream-Gruppe zu verwenden. Sie können dieses Feld jederzeit bearbeiten.

Kapazität bearbeiten

Skalieren Sie Ihre Stream-Gruppen, indem Sie die Kapazität für jeden Standort anpassen.

Weitere Informationen über [Amazon GameLift Streams-Servicekontingente](#) die Kapazitätskontingente für Streamgruppen pro Standort und darüber AWS-Konto, wie Sie diese Kontingente erhöhen können, finden Sie unter.

Console

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Wählen Sie in der Navigationsleiste Stream-Gruppen aus, um eine Liste Ihrer vorhandenen Stream-Gruppen anzuzeigen. Wählen Sie die Stream-Gruppe aus, die Sie bearbeiten möchten.
3. Wählen Sie auf der Detailseite der Stream-Gruppe die Option Konfiguration bearbeiten aus.
4. Geben Sie für jeden Standort neue Werte für die ständig verfügbare Kapazität, die maximale Kapazität und die Zielkapazität im Leerlauf in die entsprechenden Zellen der Tabelle ein. Die Werte für die Kapazität müssen ein ganzzahliges Vielfaches des Tenancy-Werts der Stream-Klasse der Stream-Gruppe sein.

Wenn Sie den Wert für die Always-On-Kapazität auf Null setzen, weist die Stream-Gruppe dem Stream keine Hosts zu.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren. AWS-Region Anweisungen zur Einrichtung finden Sie unter[Laden Sie das herunter AWS CLI](#).

Um die Stream-Kapazität zu bearbeiten, verwenden Sie den AWS CLI

AWS CLI Verwenden Sie dabei den [UpdateStreamGroup](#)Befehl, der an Ihren Inhalt angepasst ist.

```
aws gameliftstreams update-stream-group \
--identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4 \
--location-configurations '[{"LocationName": "us-east-1", "AlwaysOnCapacity": 4,
"MaximumCapacity": 8}, \
{"LocationName": "ap-northeast-1", "AlwaysOnCapacity": 0,
"MaximumCapacity": 2, "TargetIdleCapacity": 1}]'
```

where

identifier

Ein [Amazon-Ressourcenname \(ARN\)](#) oder eine Amazon-ID, die die Stream-Gruppenressource eindeutig identifiziert.

ARN-Beispiel: arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4

ID-Beispiel: sg-1AB2C3De4

location-configurations

Eine Reihe von Standorten, die in dieser Stream-Gruppe mit ihren neuen Kapazitäten aktualisiert werden sollen. Bei den Kapazitätswerten muss es sich um ganzzahlige Vielfache des Tenancy-Werts der Stream-Klasse der Stream-Gruppe handeln.

Wenn Sie die Kapazität eines Standorts einer Stream-Gruppe aktualisieren, beginnt Amazon GameLift Streams mit der Bearbeitung Ihrer Anfrage, was einige Zeit dauern kann. Während dieser Zeit arbeitet Amazon GameLift Streams daran, Ressourcen in der Stream-Gruppe nach Bedarf zuzuweisen oder freizugeben, um die von Ihnen festgelegte Always-On-Stream-Kapazität zu erreichen. Sie können den Bereitstellungsstatus Ihrer Stream-Kapazität einsehen, indem Sie die Seite mit den Stream-Gruppendetails in der Amazon GameLift Streams-Konsole aufrufen oder die [GetStreamGroupAPI](#) aufrufen.

Wenn sich Ihre Stream-Gruppe im Status Aktiv befindet, Stream-Kapazität verfügbar ist und die Anwendung die Replikation an den Speicherort abgeschlossen hat, an dem Sie streamen möchten,

können Sie mit dem Streaming beginnen. Weitere Informationen finden Sie unter [Stream-Sitzungen mit Amazon GameLift Streams starten](#).

Verhalten beim Herunterskalieren der Kapazität

Wenn Sie die Kapazität herunterskalieren, wartet Amazon GameLift Streams, bis der Host inaktiv ist, bevor er freigegeben wird. Da ein Host 1 oder 2 Sitzungen unterstützen kann, ist der Host nur inaktiv, wenn alle aktiven Sitzungen auf dem Host enden. Eine Stream-Sitzung endet, wenn der Benutzer seine Sitzung beendet oder die Sitzung das Zeitlimit überschreitet. In Extremsituationen, in denen bestehende Sitzungen die maximal mögliche Dauer erreichen dürfen, kann es daher bis zu 24 Stunden dauern, bis die gewünschte Kapazität erreicht ist. Wenn Sie das Ende aller aktiven Stream-Sitzungen in einer Stream-Gruppe erzwingen möchten, können Sie die Stream-Gruppe in der Konsole oder mithilfe der [DeleteStreamGroup](#)API löschen, oder Sie können die [TerminateStreamSession](#)API verwenden, um aktive Sitzungen nacheinander zu beenden.

Fügen Sie Standorte zu einer Stream-Gruppe hinzu

Console

So fügen Sie Standorte mithilfe der Amazon GameLift Streams-Konsole zu einer Stream-Gruppe hinzu

1. Wählen Sie in der Navigationsleiste Stream-Gruppen aus, um eine Liste Ihrer vorhandenen Stream-Gruppen anzuzeigen. Wählen Sie die Stream-Gruppe aus, zu der Sie neue Standorte hinzufügen möchten.
2. Wählen Sie auf der Seite mit den Stream-Gruppendetails die Option Konfiguration bearbeiten aus.
3. Aktivieren Sie das Kontrollkästchen neben den Standorten, die Sie zu dieser Stream-Gruppe hinzufügen möchten, und legen Sie dann deren Kapazitäten fest.
4. Sehen Sie sich die Zusammenfassung Ihrer ausgewählten Standorte an, einschließlich der Kosten für die Stream-Kapazität. Wählen Sie Speichern, um Ihre Auswahl zu bestätigen.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren AWS-Region. Anweisungen zur Einrichtung finden Sie unter [Laden Sie das herunter AWS CLI.](#)

Um Standorte zu einer Stream-Gruppe hinzuzufügen, verwenden Sie den AWS CLI

AWS CLI Verwenden Sie dabei den [AddStreamGroupLocations](#)Befehl, der auf Ihren Inhalt zugeschnitten ist.

```
aws gameliftstreams add-stream-group-locations \
--identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4
--location-configurations '[{"LocationName": "us-east-1", "AlwaysOnCapacity": 2,
"MaximumCapacity": 4, "TargetIdleCapacity": 1}]'
```

where

identifier

Ein [Amazon-Ressourcenname \(ARN\)](#) oder eine Amazon-ID, die die Stream-Gruppenressource eindeutig identifiziert.

ARN-Beispiel: arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/ sg-1AB2C3De4

ID-Beispiel: sg-1AB2C3De4

location-configurations

Eine Reihe von Standorten, die dieser Stream-Gruppe hinzugefügt werden sollen, und deren Kapazitäten. Eine vollständige Liste der Standorte, die Amazon GameLift Streams unterstützt, finden Sie unter [AWS-Regionen und entfernte Standorte, die von Amazon GameLift Streams unterstützt werden.](#)

Die Kapazitätswerte müssen ein ganzzahliges Vielfaches des Tenancy-Werts der Stream-Klasse der Stream-Gruppe sein.

Wenn Ihre Anwendung die Replikation an die neuen Speicherorte abgeschlossen hat und Ihre Stream-Gruppe über verfügbare Stream-Kapazität verfügt, können Sie mit dem Streaming von den neuen Speicherorten aus beginnen. Weitere Informationen zum Streamen finden Sie unter.

[Stream-Sitzungen mit Amazon GameLift Streams starten](#) Amazon GameLift Streams beginnt mit der Bearbeitung Ihrer Anfrage. Während dieser Zeit repliziert Amazon GameLift Streams Ihre Anwendung und weist Rechenressourcen an den neuen Standorten zu. Sie können den Status der Replikation im Abschnitt Verknüpfte Anwendungen der Seite mit den Stream-Gruppendetails einsehen, indem Sie in der Spalte Replikationsstatus auf den Status klicken.

Standorte in einer Stream-Gruppe entfernen

Um Rechenressourcen von bestimmten Standorten nicht mehr zu verwenden, können Sie die Standorte aus Ihrer Stream-Gruppe entfernen. Sie können den primären Standort einer Stream-Gruppe nicht entfernen. Wenn Sie jedoch keine Rechenressourcen an diesem Standort benötigen, können Sie die Stream-Kapazitäten auf Null setzen.

Warning

Wenn Sie einen Standort in einer Stream-Gruppe entfernen, trennt Amazon GameLift Streams die aktiven Streams an diesem Standort, wodurch der Stream aller verbundenen Endbenutzer gestoppt wird.

Console

So entfernen Sie Standorte mithilfe der Amazon GameLift Streams-Konsole aus einer Stream-Gruppe

1. Wählen Sie im Navigationsbereich Stream-Gruppen aus, um eine Liste Ihrer vorhandenen Stream-Gruppen anzuzeigen.
2. Wählen Sie den Namen der Stream-Gruppe aus, aus der Sie Standorte entfernen möchten.
3. Wählen Sie auf der Seite mit den Stream-Gruppendetails die Option Konfiguration bearbeiten aus.
4. Deaktivieren Sie das Kontrollkästchen neben dem Namen des Speicherorts, den Sie entfernen möchten.
5. Wählen Sie Speichern.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten AWS-Region konfigurieren. Anweisungen zur Einrichtung finden Sie unter [Laden Sie das herunter AWS CLI.](#)

Um Standorte aus einer Stream-Gruppe zu entfernen, verwenden Sie den AWS CLI

AWS CLI Verwenden Sie dabei den [RemoveStreamGroupLocations](#)Befehl, der auf Ihren Inhalt zugeschnitten ist.

```
aws gameliftstreams remove-stream-group-locations \
--identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4
--locations us-east-1 eu-central-1
```

where

identifier

Ein [Amazon-Ressourcenname \(ARN\)](#) oder eine Amazon-ID, die die Stream-Gruppenressource eindeutig identifiziert.

ARN-Beispiel: arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4

ID-Beispiel: sg-1AB2C3De4

locations

Eine Reihe von Speicherorten, die aus dieser Stream-Gruppe entfernt werden sollen. Eine vollständige Liste der Standorte, die Amazon GameLift Streams unterstützt, finden Sie unter [AWS-Regionen und entfernte Standorte, die von Amazon GameLift Streams unterstützt werden.](#)

Löschen Sie eine Stream-Gruppe

Sie können eine Stream-Gruppe löschen, die sich in einem beliebigen Status befindet. Diese Aktion löscht die Stream-Gruppe dauerhaft und gibt ihre Rechenressourcen frei. Wenn Streams in Bearbeitung sind, werden sie durch diese Aktion gestoppt und Ihre Endbenutzer können den Stream nicht mehr ansehen.

Es hat sich bewährt, vor dem Löschen einer Stream-Gruppe zu überprüfen, ob Streams in Bearbeitung sind, und Maßnahmen zu ergreifen, um sie zu stoppen.

Console

Um eine Stream-Gruppe mit der Amazon GameLift Streams-Konsole zu löschen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Um eine Liste Ihrer vorhandenen Stream-Gruppen anzuzeigen, wählen Sie im Navigationsbereich Stream-Gruppen aus.
3. Wählen Sie den Namen der Stream-Gruppe aus, die Sie löschen möchten.
4. Wählen Sie auf der Detailseite der Stream-Gruppe die Option Löschen aus.
5. Bestätigen Sie im Dialogfeld Löschen die Löschaktion.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren AWS-Region. Anweisungen zur Einrichtung finden Sie unter[Laden Sie das herunter AWS CLI](#).

Um deine Stream-Gruppe mit dem zu löschen AWS CLI

AWS CLI Verwenden Sie dabei den [DeleteStreamGroup](#)Befehl, der auf Ihren Inhalt zugeschnitten ist.

```
aws gameliftstreams delete-stream-group \
--identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4
```

where

identifier

Ein [Amazon-Ressourcenname \(ARN\)](#) oder eine Amazon-ID, die die Stream-Gruppenressource eindeutig identifiziert.

ARN-Beispiel: arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4

ID-Beispiel: sg-1AB2C3De4

Amazon GameLift Streams beginnt mit der Freigabe von Rechenressourcen und dem Löschen der Stream-Gruppe. Während dieser Zeit befindet sich die Stream-Gruppe im Status Löschen. Nachdem Amazon GameLift Streams die Stream-Gruppe gelöscht hat, können Sie sie nicht mehr abrufen.

Verknüpfte Anwendungen

Wenn Sie mehrere Anwendungen streamen möchten, die denselben Pool von Rechenressourcen verwenden, können Sie mehrere Anwendungen mit derselben Stream-Gruppe verknüpfen. Ebenso können Sie eine Anwendung mit mehreren Stream-Gruppen verknüpfen, wenn Sie eine Anwendung mit unterschiedlichen Gruppen von Rechenressourcen streamen möchten.

Weitere Informationen zum Verknüpfen von Anwendungen mit Stream-Gruppen finden Sie unter [Übersicht über Stream-Gruppen für mehrere Anwendungen](#).

Lebenszyklus einer Stream-Gruppe

Stream-Gruppen haben eine maximale Lebensdauer von 365 Tagen. Als bewährte Methode empfehlen wir, Stream-Gruppen alle 3-4 Wochen neu zu erstellen, um wichtige Service-Updates und Fixes zu erhalten und eine optimale Leistung sicherzustellen. Das Neuerstellen einer Stream-Gruppe hat keine Auswirkungen auf Ihre hochgeladenen Anwendungen.

Mit zunehmendem Alter von Stream-Gruppen gelten die folgenden Einschränkungen:

- Nach 180 Tagen: Sie können die Stream-Gruppe nicht mehr mit neuen Anwendungszuordnungen aktualisieren
- Nach 365 Tagen: Die Stream-Gruppe läuft ab und es können keine Sessions mehr gestreamt werden

Das mit der Stream-Gruppe verknüpfte Konto erhält zwei Erinnerungsbenachrichtigungen von AWS Health: eine am 45-Tag und eine zweite Erinnerung am 150-Tag. Diese Benachrichtigungen erinnern Sie daran, dass die Funktion zur Anwendungszuweisung am 180-Tag verloren geht. Außerdem erhalten Sie am Tag 335 eine letzte Benachrichtigung, die Sie daran erinnert, dass Stream-Gruppen am 365-Tag ablaufen. Wartungswarnungen werden auch im AWS Health Dashboard und auf den Seiten der Stream-Gruppen in der Amazon GameLift Streams-Konsole angezeigt.

Das Ablaufdatum einer Stream-Gruppe finden Sie auf der Seite mit den Stream-Gruppendetails in der Konsole oder verwenden Sie das `ExpiresAt` Feld in der [GetStreamGroup](#)API-Antwort.

Eine abgelaufene Stream-Gruppe hat den Status EXPIRED und ist schreibgeschützt. Sie können sie nicht aktualisieren oder neue Stream-Sitzungen starten. Um die Funktionalität wiederherzustellen, erstellen Sie die Stream-Gruppe neu.

Wartung der Stream-Gruppe

Immer wenn eine Funktion veröffentlicht wird, für deren Verwendung eine neue Stream-Gruppe erforderlich ist, wird oben auf der Detailseite der Stream-Gruppe die Meldung „Wartung erforderlich“ angezeigt, die Sie darüber informiert, dass sie veraltet ist. Das Neuerstellen einer Stream-Gruppe ist ein manueller Vorgang. Um Ihnen dabei zu helfen, verwenden Sie die Schaltfläche „Stream-Gruppe erstellen“ in der Nachricht, um den Vorgang zu starten. Einige der Felder werden für Sie ausgefüllt.

Die Wartung der Stream-Gruppe ist auch erforderlich, wenn die Stream-Gruppe älter als 180 Tage ist. Sie können keine neuen Anwendungen mehr mit diesen älteren Stream-Gruppen verknüpfen, bis sie neu erstellt wurden. Nach 365 Tagen ist das Streamen über die Stream-Gruppe nicht mehr möglich und Änderungen an der Stream-Gruppe sind nicht mehr zulässig.

Übersicht über Stream-Gruppen für mehrere Anwendungen

Eine Stream-Gruppe mit mehreren Anwendungen ist eine Stream-Gruppe, die mit mehreren Anwendungen verknüpft ist. Auf diese Weise können Sie mehrere Anwendungen streamen, indem Sie denselben Satz von Rechenressourcen in einer einzigen Stream-Gruppe verwenden.

Ein häufiger Anwendungsfall für Stream-Gruppen mit mehreren Anwendungen ist die Veröffentlichung verschiedener Versionen Ihres Spiels. Nehmen wir zum Beispiel an, Sie haben eine Stream-Gruppe erstellt und als Standardanwendung die Originalversion Ihres Spiels festgelegt. Nehmen wir dann an, Sie erstellen zusätzliche Anwendungen, die andere Versionen Ihres Spiels enthalten, und verknüpfen diese mit der Stream-Gruppe. Da diese Anwendungen derselben Stream-Gruppe zugeordnet sind, müssen Sie nur einen einzigen Satz von Rechenressourcen oder Stream-Kapazität verwalten, um all diese Spiele zu streamen. Das bedeutet, dass unabhängig davon, welche Anwendung ein Endbenutzer streamt, die Anwendung auf einer Rechenressource aus derselben Menge ausgeführt wird, die dieser Stream-Gruppe zugewiesen wurde.

Hier sind weitere mögliche Beispiele aus der Praxis:

- Eine Game-Streaming-Plattform, die Kunden verschiedene Streaming-Stufen bietet.

- Ein Qualitätssicherungsteam, das mehrere Versionen eines Spiels testet.
- Um das Stream-Kapazitätsmanagement zu vereinfachen, indem eine einzelne Stream-Gruppe für mehrere Anwendungen verwendet wird.
- Um mehreren Anwendungen das Streamen aus demselben Stream-Kapazitätspool zu ermöglichen.

Einschränkungen und Anforderungen

Sie können Anwendungen nur Streamgruppen zuordnen, die über kompatible Laufzeitumgebungen und Stream-Klassen verfügen. Weitere Informationen finden Sie unter [Kompatibilität mit Laufzeitumgebung und Stream-Klassen](#).

Die folgenden Zuordnungsbeschränkungen gelten für Anwendungen und Stream-Gruppen. Diese Grenzwerte sind im Service für alle Kunden festgelegt.

Name	Standard	Anpassbar	Description
Anwendungen in einer Stream-Gruppe	250	Nein	Die maximale Anzahl von Amazon GameLift Streams-Anwendungen, die einer Stream-Gruppe zugeordnet werden können.
Stream-Gruppenzuordnungen pro Anwendung	100	Nein	Die maximale Anzahl von Stream-Gruppen, denen eine Amazon GameLift Streams-Anwendung zugeordnet werden kann.

Über Standardanwendungen

Jede Stream-Gruppe hat eine Standardanwendung, die zunächst die erste Anwendung ist, die Sie der Stream-Gruppe hinzufügen. Die Standardanwendung wird automatisch auf allen Always-On-Computing-Ressourcen vorab zwischengespeichert, was dazu beitragen kann, die Ladezeit der

Anwendung beim Start des Streams zu reduzieren. Der Amazon GameLift Streams-Service kann während seiner Optimierungsprozesse auch andere verknüpfte Anwendungen zwischenspeichern.

Eigenschaften von Standardanwendungen und anderen verknüpften Anwendungen:

- Die Standardanwendung wird vorab zwischengespeichert (auf vorab zugewiesenen Rechenressourcen wie Ihrer ständig verfügbaren Kapazität), um die Ladezeit der Anwendung beim Stream-Start zu reduzieren.
- Die Standardanwendung kann geändert werden. Beachten Sie, dass es beim Wechseln zwischen Standardanwendungen in einer Stream-Gruppe bis zu einigen Stunden dauern kann, bis die neue Standardanwendung an allen Speicherorten vorab zwischengespeichert ist.
- Es ist mindestens eine verknüpfte Anwendung erforderlich, bevor Sie mit dem Streaming aus der Stream-Gruppe beginnen können. Die erste verknüpfte Anwendung wird automatisch zur Standardanwendung gemacht.
- Wenn Sie die Verknüpfung der Standardanwendung einer Stream-Gruppe aufheben, wählt Amazon GameLift Streams automatisch eine neue Standardanwendung aus den verbleibenden zugehörigen Anwendungen, falls vorhanden.
- Dieselbe Anwendung kann die Standardanwendung für mehrere Stream-Gruppen sein.
- Die Gruppe der verknüpften Anwendungen ist veränderbar, bis die Stream-Gruppe 180 Tage alt ist. In der Praxis bedeutet dies, dass Sie Anwendungen verknüpfen und die Verknüpfung aufheben können, bis die Stream-Gruppe 180 Tage alt ist. Danach können Sie die Verknüpfung von Anwendungen mit einer Stream-Gruppe nur noch für den Rest des Lebenszyklus der Stream-Gruppe trennen.

Standardanwendung ändern

Wenn Sie die erste Anwendung mit einer Stream-Gruppe verknüpfen, wird sie automatisch zur Standardanwendung und erhält Vorteile vor dem Zwischenspeichern. Sie können die Standardanwendung jederzeit ändern, um diese Vorteile einer anderen Anwendung zuzuweisen.

Note

Wenn Sie zwischen Standardanwendungen in einer Stream-Gruppe wechseln, kann es bis zu einigen Stunden dauern, bis die neue Standardanwendung an allen Speicherorten vorab zwischengespeichert ist.

Console

So ändern Sie die Standardanwendung mithilfe der Amazon GameLift Streams-Konsole

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Wählen Sie in der Navigationsleiste Stream-Gruppen aus, um eine Liste Ihrer vorhandenen Stream-Gruppen anzuzeigen.
3. Wählen Sie eine Stream-Gruppe aus, um deren Details anzuzeigen.
4. Wählen Sie unter Verknüpfte Anwendungen die Anwendung aus, die Sie zur Standardanwendung machen möchten.
5. Wählen Sie Als Standard festlegen.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren AWS-Region. Anweisungen zur Einrichtung finden Sie unter[Laden Sie das herunter AWS CLI](#).

Um die Standardanwendung mit dem zu ändern AWS CLI

AWS CLI Verwenden Sie dabei den [UpdateStreamGroup](#)Befehl, der an Ihren Inhalt angepasst ist. Die Anwendung, die Sie zur Standardanwendung machen möchten, muss bereits mit der Stream-Gruppe verknüpft sein.

```
aws gameliftstreams update-stream-group \
--identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4 \
--default-application-identifier a-9ZY8X7Wv6
```

where

- **identifier:**

Eine Stream-Gruppe mit einer Anwendung, die Sie zur Standardanwendung machen möchten.

Bei diesem Wert kann es sich um einen [Amazon-Ressourcennamen \(ARN\)](#) oder eine Amazon-ID handeln, die die Stream-Gruppenressource eindeutig identifiziert.

ARN-Beispiel: `arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4`

ID-Beispiel: `sg-1AB2C3De4`

- `default-application-identifier`:

Die Anwendung, die Sie zur Standardanwendung in dieser Stream-Gruppe machen möchten.

Dieser Wert ist ein [Amazon-Ressourcename \(ARN\)](#) oder eine Amazon-ID, die die Anwendungsressource eindeutig identifiziert.

ARN-Beispiel: `arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6`

ID-Beispiel: `a-9ZY8X7Wv6`

Eine Anwendung verknüpfen

Wenn Sie eine Anwendung mit einer Stream-Gruppe verknüpfen oder ihr zuordnen, kann die Stream-Gruppe die Anwendung streamen. Wenn es sich um die erste Anwendung in der Stream-Gruppe handelt, wird sie automatisch zur Standardanwendung. Sie können weitere Anwendungen mit einer Stream-Gruppe verknüpfen oder deren Verknüpfung aufheben, bis sie 180 Tage alt ist. Danach können Sie die Verknüpfung von Anwendungen mit einer Stream-Gruppe nur noch für den Rest des Lebenszyklus der Gruppe trennen.

Important

Sie können eine Anwendung nicht mit einer Stream-Gruppe verknüpfen, die älter als 180 Tage ist. Um der Stream-Gruppe verschiedene Anwendungen zuzuordnen, müssen Sie sie zunächst neu erstellen. Anweisungen zum Neuerstellen einer Stream-Gruppe finden Sie unter. [Wartung der Stream-Gruppe](#)

Bevor Sie eine Anwendung verknüpfen, stellen Sie sicher, dass sich die Stream-Gruppe im Status Aktiv befindet.

Console

So verlinken Sie über die Amazon GameLift Streams-Konsole

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Wählen Sie in der Navigationsleiste Stream-Gruppen aus, um eine Liste Ihrer vorhandenen Stream-Gruppen anzuzeigen.
3. Wählen Sie eine Stream-Gruppe aus, um deren Details anzuzeigen.
4. Wählen Sie unter Verknüpfte Anwendungen die Option Anwendung verknüpfen aus.
5. Wählen Sie eine Anwendung aus, die Sie verknüpfen möchten. Bestätigen Sie Ihre Auswahl und wählen Sie Anwendung verknüpfen.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren AWS-Region. Anweisungen zur Einrichtung finden Sie unter[Laden Sie das herunter AWS CLI](#).

So verknüpfen Sie eine (n) Anwendung (en) mit dem AWS CLI

AWS CLI Verwenden Sie dabei den [AssociateApplications](#)Befehl, der auf Ihren Inhalt zugeschnitten ist.

```
aws gameliftstreams associate-applications \
--identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4 \
--application-identifiers a-9ZY8X7Wv6 a-1Z78C7Wv6
```

where

- **identifier:**

Eine Stream-Gruppe, mit der diese Anwendungen verknüpft werden sollen.

Bei diesem Wert kann es sich um einen [Amazon-Ressourcennamen \(ARN\)](#) oder eine Amazon-ID handeln, die die Stream-Gruppenressource eindeutig identifiziert.

ARN-Beispiel: `arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4`

ID-Beispiel: `sg-1AB2C3De4`

- **application-identifiers:**

Eine Reihe von Anwendungen, die Sie mit dieser Stream-Gruppe verknüpfen möchten.

Dieser Wert ist ein [Amazon-Ressourcename \(ARN\)](#) oder eine Amazon-ID, die die Anwendungsressource eindeutig identifiziert.

ARN-Beispiel: `arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6`

ID-Beispiel: `a-9ZY8X7Wv6`

Verknüpfung mit einer Anwendung aufheben

Wenn Sie die Verknüpfung einer Anwendung zu einer Stream-Gruppe aufheben oder deren Zuordnung aufheben, können Sie diese Anwendung nicht mehr streamen, indem Sie die der Stream-Gruppe zugewiesenen Rechenressourcen verwenden. Alle in Bearbeitung befindlichen Streams werden fortgesetzt, bis sie beendet werden, wodurch vermieden wird, dass der Stream eines Endbenutzers unterbrochen wird. Amazon GameLift Streams initiiert keine neuen Streams mit dieser Stream-Gruppe. Die Aktion „Verbindung aufheben“ hat keinen Einfluss auf die Stream-Kapazität einer Stream-Gruppe.

Wenn Sie die Verknüpfung der Standardanwendung einer Stream-Gruppe aufheben, wählt Amazon GameLift Streams automatisch eine neue Standardanwendung aus den verbleibenden zugehörigen Anwendungen, falls vorhanden.

Console

So heben Sie die Verknüpfung mit der Amazon GameLift Streams-Konsole auf

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Wählen Sie in der Navigationsleiste Stream-Gruppen aus, um eine Liste Ihrer vorhandenen Stream-Gruppen anzuzeigen.

3. Wählen Sie eine Stream-Gruppe aus, um deren Details anzuzeigen.
4. Wählen Sie unter Verknüpfte Anwendungen die Anwendung (en) aus, deren Verknüpfung Sie aufheben möchten. Wählen Sie „Anwendungsverknüpfung aufheben“.
5. Bestätigen Sie im Dialogfeld „Anwendungsverknüpfung aufheben“ die Aktion „Verknüpfung aufheben“ und wählen Sie „Verknüpfung aufheben“.

CLI

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren. AWS-Region Anweisungen zur Einrichtung finden Sie unter[Laden Sie das herunter AWS CLI.](#)

Um die Verknüpfung einer Anwendung (en) aufzuheben, verwenden Sie AWS CLI

AWS CLI Verwenden Sie dabei den [DisassociateApplications](#)Befehl, der auf Ihren Inhalt zugeschnitten ist.

```
aws gameliftstreams disassociate-applications \
--identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4 \
--application-identifiers a-9ZY8X7Wv6 a-1Z78C7Wv6
```

where

- **identifier:**

Eine Stream-Gruppe, mit der die Verknüpfung dieser Anwendungen aufgehoben werden soll.

Bei diesem Wert kann es sich um einen [Amazon-Ressourcennamen \(ARN\)](#) oder eine Amazon-ID handeln, die die Stream-Gruppenressource eindeutig identifiziert.

ARN-Beispiel: arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4

ID-Beispiel: sg-1AB2C3De4

- **application-identifiers:**

Eine Reihe von Anwendungen, deren Verknüpfung mit dieser Stream-Gruppe aufgehoben werden soll.

Dieser Wert ist ein [Amazon-Ressourcenname \(ARN\)](#) oder eine Amazon-ID, die die Anwendungsressource eindeutig identifiziert.

ARN-Beispiel: `arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6`

ID-Beispiel: `a-9ZY8X7Wv6`

Stream-Sitzungen mit Amazon GameLift Streams starten

In diesem Abschnitt werden Stream-Sessions behandelt, also die eigentliche Instanz eines Streams, bei der ein Endbenutzer oder Spieler mit Ihrer Anwendung interagieren oder Ihr Spiel spielen kann. Sie erfahren, wie Sie Ihre eigene Stream-Sitzung testen und den Lebenszyklus einer Stream-Sitzung verstehen.

Um Stream-Sitzungen für Endbenutzer zu starten, müssen Sie Amazon GameLift Streams in Ihren eigenen Service integrieren. Weitere Informationen finden Sie unter [Backend-Service und WebClient von Amazon GameLift Streams](#).

Über Stream-Sitzungen

Die Voraussetzungen für den Start einer Stream-Sitzung sind eine Anwendung im Status Bereit, eine Stream-Gruppe mit verfügbarer Kapazität an dem Ort, an dem Sie streamen möchten, und die Anwendung, die an den Speicherort repliziert wurde, an dem Sie streamen möchten. Eine Stream-Sitzung wird auf einer der Rechenressourcen ausgeführt, die einer Stream-Gruppe zugewiesen wurden. Wenn Sie einen Stream starten, müssen Sie eine Stream-Gruppe und eine Anwendung angeben, die mit ihren ARN- oder ID-Werten gestreamt werden soll.

Wenn Sie eine Stream-Sitzung erfolgreich starten, erhalten Sie eine eindeutige Kennung für diese Stream-Sitzung. Anschließend verwenden Sie diese ID, um die Stream-Sitzung mit einem Endbenutzer zu verbinden. Weitere Informationen finden Sie [StartStreamSession](#) in der Amazon GameLift Streams-API-Referenz.

Einen Stream in der Konsole testen

Am direktesten können Sie über die Amazon Streams-Konsole testen, wie Ihre Anwendung GameLift streamt. Wenn Sie einen Stream starten, verwendet Amazon GameLift Streams eine der Rechenressourcen, die Ihre Stream-Gruppe zuweist. Sie müssen also über verfügbare Kapazität in Ihrer Stream-Gruppe verfügen.

Um Ihren Stream in der Amazon GameLift Streams-Konsole zu testen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Sie können einen Stream auf verschiedene Arten testen. Beginne auf der Seite Stream-Gruppen oder Stream testen und folge diesen Schritten:
 - a. Wählen Sie eine Stream-Gruppe aus, die Sie zum Streamen verwenden möchten.
 - b. Wenn du auf der Seite Stream-Gruppen startest, wähle Stream testen. Wenn du auf der Seite „Stream testen“ startest, wähle Wählen aus. Dadurch wird die Seite „Stream-Konfiguration testen“ für die ausgewählte Stream-Gruppe geöffnet.
 - c. Wählen Sie unter Verknüpfte Anwendungen eine Anwendung aus.
 - d. Wählen Sie unter Standort einen Standort mit verfügbarer Kapazität aus.
 - e. (Optional) Geben Sie unter Programmkonfigurationen Befehlszeilenargumente oder Umgebungsvariablen ein, die beim Start an die Anwendung übergeben werden sollen.
 - f. Bestätigen Sie Ihre Auswahl und wählen Sie Stream testen aus.
3. Nachdem dein Stream geladen wurde, kannst du die folgenden Aktionen in deinem Stream ausführen:
 - a. Um Eingaben wie Maus, Tastatur und Gamepad zu verbinden (mit Ausnahme von Mikrofonen, die im Teststream nicht unterstützt werden), wähle Eingabe anhängen. Du verbindest deine Maus automatisch, wenn du den Cursor in das Stream-Fenster bewegst.
 - b. Um Dateien, die während der Streaming-Sitzung erstellt wurden, am Ende der Sitzung in einen Amazon S3 S3-Bucket exportieren zu lassen, wählen Sie Dateien exportieren und geben Sie die Bucket-Details an. Exportierte Dateien finden Sie auf der Seite Sessions.
 - c. Um den Stream im Vollbildmodus anzusehen, wählen Sie Vollbild. Drücken Sie Esc, um diese Aktion rückgängig zu machen.

- Um den Stream zu beenden, wählen Sie Sitzung beenden. Wenn die Verbindung zum Stream unterbrochen wird, steht die Stream-Kapazität zum Starten eines weiteren Streams zur Verfügung.

 Note

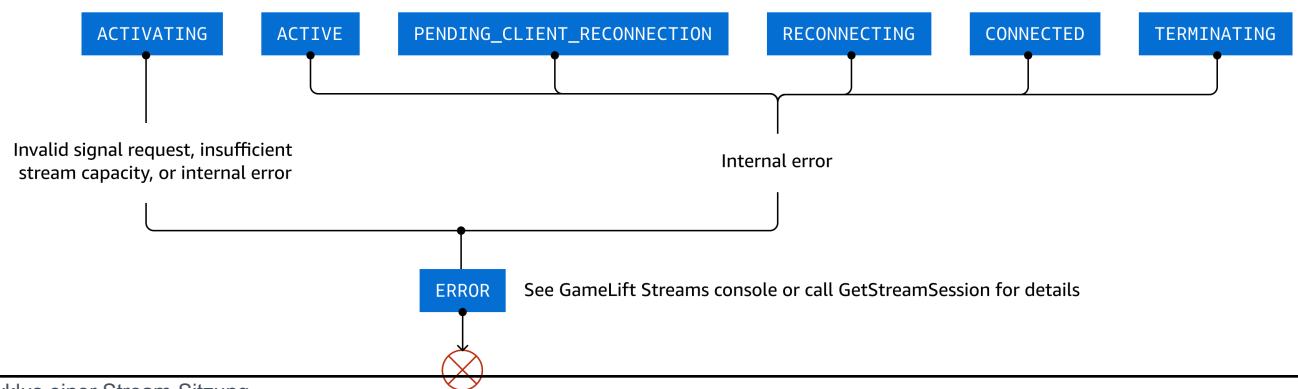
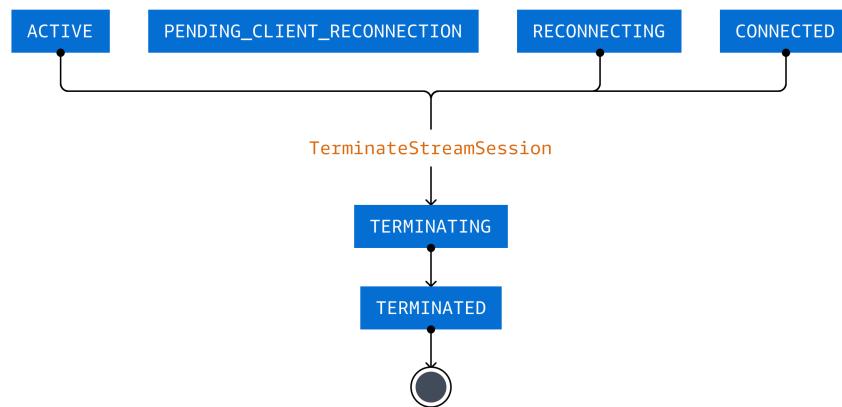
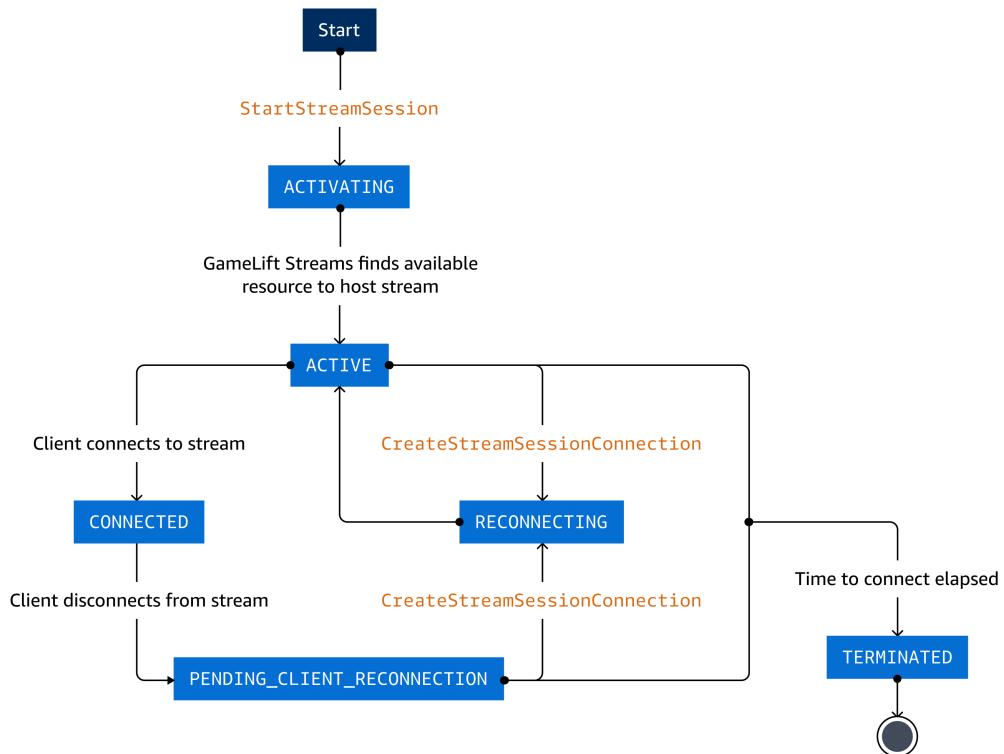
Die Funktion Stream testen in der Amazon GameLift Streams-Konsole unterstützt keine Mikrofone.

Lebenszyklus einer Stream-Sitzung

Wenn Sie mit Stream-Sitzungen in Amazon GameLift Streams arbeiten, kann Ihnen dieses Diagramm helfen, die verschiedenen Zustände zu verstehen, in die eine Stream-Sitzung während ihres gesamten Lebenszyklus übergeht.

- [StartStreamSession](#) erstellt eine neue Stream-Sitzung, die im ACTIVATING Status beginnt. Wenn Amazon GameLift Streams verfügbare Ressourcen zum Hosten des Streams findet, wechselt die Stream-Sitzung zu ACTIVE. Wenn ein Client eine Verbindung zum aktiven Stream herstellt, wechselt die Stream-Sitzung zu CONNECTED.
- Wenn ein Client die Verbindung zu einem Stream trennt, geht die Stream-Sitzung in den PENDING_CLIENT_RECONNECTION Status über. [CreateStreamSessionConnection](#) wechselt die Stream-Sitzung in den Stream und veranlasst entweder den ClientRECONNECTING, erneut eine Verbindung zum Stream herzustellen, oder erstellt eine neue Stream-Sitzung. Wenn eine Stream-Sitzung bereit ist, damit der Client wieder eine Verbindung herstellen kann, wechselt sie zu ACTIVE. Wenn der Client wieder eine Verbindung herstellt, wechselt er zurück zu CONNECTED. Wenn die Verbindung zu einem Client länger als unterbrochen wird `ConnectionTimeoutSeconds`, endet die Stream-Sitzung.
- Wenn ein Client innerhalb des von angegebenen Zeitraums keine Verbindung zu einer Stream-Sitzung in ACTIVE oder PENDING_CLIENT_RECONNECTION Status herstellt `ConnectionTimeoutSeconds`, wechselt er zu TERMINATED.
- [TerminateStreamSession](#) leitet die Beendigung des Streams ein und die Stream-Sitzung wechselt in den TERMINATING Status. Wenn die Stream-Sitzung erfolgreich beendet wurde, wechselt sie zu TERMINATED.
- Eine Stream-Sitzung in einem beliebigen Zustand, außer TERMINATED, kann zu ERROR wechseln. Wenn ein API-Aufruf ERROR als Statuswert zurückgegeben wird, überprüfen Sie den Wert von,

StatusReason um eine kurze Beschreibung der Fehlerursache zu erhalten. Sie können diese Werte auch aufrufen [GetStreamSession](#), um diese Werte zu überprüfen.



Timeout-Werte, die sich auf Stream-Sitzungen auswirken

Stream-Sitzungen werden durch mehrere Timeout-Werte gesteuert, die verschiedene Aspekte des Sitzungslebenszyklus steuern. In etwa chronologischer Reihenfolge, in der sie normalerweise während des Lebenszyklus einer Stream-Sitzung auftreten können, beinhalten sie Folgendes:

Timeout bei der Platzierung

Zeitlimit für Amazon GameLift Streams, um Rechenressourcen für das Hosten einer Stream-Sitzung mit verfügbarer Kapazität zu finden. Das Zeitlimit für die Platzierung hängt vom Kapazitätstyp ab, der zur Erfüllung Ihrer Stream-Anfrage verwendet wurde:

- Always-On-Kapazität: 75 Sekunden
- Kapazität auf Abruf:
 - Linux/Proton-Laufzeiten: 90 Sekunden
 - Windows-Laufzeit: 10 Minuten
- Verhalten: Wenn Amazon GameLift Streams innerhalb dieser Zeit keine verfügbaren Ressourcen identifizieren kann, Status wechselt die Stream-Sitzung zu ERROR einem StatusReason vonplacementTimeout.

Verbindungs-Timeout

Zeitspanne, in der Amazon GameLift Streams darauf wartet, dass ein Client eine Verbindung zu einer Stream-Sitzung herstellt oder erneut eine Verbindung herstellt.

- Parameter: in ConnectionTimeoutSeconds [StartStreamSession](#)
- Bereich: 1 - 3600 Sekunden (1 Stunde)
- Standard: 120 Sekunden (2 Minuten)
- Verhalten: Der Timer startet, wenn die Stream-Sitzung den PENDING_CLIENT_RECONNECTION Status ACTIVE Or erreicht. Wenn vor dem Timeout kein Client eine Verbindung herstellt, wechselt die Sitzung Status zu TERMINATED.

Timeout bei Leerlauf

Zeitraum, in dem eine Stream-Sitzung ohne Benutzereingabe inaktiv sein kann.

- Parameter: (intern)
- Wert: 60 Minuten
- Verhalten: Wenn während dieser Zeit keine Benutzereingaben eingehen, beendet Amazon GameLift Streams die Stream-Sitzung.

Timeout für die Dauer der Sitzung

Maximale Dauer, für die Amazon GameLift Streams eine Stream-Sitzung geöffnet hält.

- Parameter: `SessionLengthSeconds` in [StartStreamSession](#)
- Bereich: 1 — 86400 Sekunden (24 Stunden)
- Standard: 43200 Sekunden (12 Stunden)
- Verhalten: Beendet die Stream-Sitzung unabhängig von einer bestehenden Client-Verbindung, wenn das Zeitlimit erreicht ist.

Beenden einer Stream-Sitzung

Wenn Sie das Beenden einer Stream-Sitzung erzwingen müssen, haben Sie die folgenden Optionen:

- Verwenden Sie die `TerminateStreamSession` API: Zur Verwendung [TerminateStreamSession](#) benötigen Sie die Stream-Gruppen-ID und die Stream-Sitzungs-ID. Sie können [ListStreamSessions](#) oder [ListStreamSessionsByAccount](#) mit dem `--status CONNECTED` Parameter verwenden, um eine Liste der Stream-Sitzungen abzurufen, mit denen ein Client verbunden ist.
- Den Standort der Sitzung aus ihrer Stream-Gruppe entfernen: Wenn Sie den Standort aus der Stream-Gruppe entfernen, in der die Sitzung gestreamt wird, werden alle aktiven Stream-Sitzungen an diesem Standort beendet. Sie können einen Standort in einer Stream-Gruppe von der Konsole oder mithilfe der [RemoveStreamGroupLocations](#) API entfernen.
- Löschen Sie die Stream-Gruppe der Sitzung: Durch das Löschen einer Stream-Gruppe werden alle aktiven Stream-Sitzungen an allen Standorten der Stream-Gruppe beendet. Sie können eine Stream-Gruppe über die Konsole oder mithilfe der [DeleteStreamGroup](#) API löschen. Gehen Sie vorsichtig vor, da Sie dadurch die Client-Verbindungen abrupt beenden werden.

Erneute Verbindung zu einer Stream-Sitzung herstellen

Wenn ein Client von einer Stream-Sitzung getrennt wird, ohne die Sitzung zu beenden, kann er innerhalb der Zeit, die `ConnectionTimeoutSeconds` beim Start der Stream-Sitzung festgelegt wurde, wieder eine Verbindung zur Sitzung herstellen. Um die Verbindung zu einer Sitzung wiederherzustellen, benötigen Sie die ID der Stream-Sitzung. Einzelheiten finden Sie [CreateStreamSessionConnection](#) in der Amazon GameLift Streams-API-Referenz. Ein Beispiel für die Wiederverbindung mit einer Stream-Sitzung finden Sie im [React Starter-Beispiel](#).

Stream-Sitzungsdateien exportieren

Während einer Stream-Sitzung kann Ihre Anwendung Ausgabedateien generieren, die Ihnen beim Debuggen oder Überprüfen Ihrer Anwendung helfen, z. B. Protokolle, Diagnoseinformationen, Absturzabbilder, Speicherdateien, Benutzerdaten und Screenshots. Die Funktion Stream-Sitzungsdateien exportieren sammelt Dateien, die während einer Sitzung erstellt oder geändert wurden, und exportiert sie als komprimierte ZIP-Datei an einen angegebenen Amazon S3 S3-Speicherort. Die Funktion sammelt außerdem jede Sekunde Leistungsstatistiken für die Sitzung, die in der ZIP-Exportdatei enthalten sind.

Warning

Bevor Sie Dateien exportieren, sollten Sie sich der folgenden Dinge bewusst sein:

- Dateien können vertrauliche Informationen enthalten, die von Ihrer Anwendung geschrieben wurden, einschließlich Anmeldeinformationen.
- Die Dateigrößen können je nach Größe Ihrer Anwendung groß sein, was sich auf Ihre Amazon S3 S3-Speicherkosten auswirkt.
- Wenn Sie einen Amazon S3 S3-Bucket in einem anderen AWS-Region auswählen, der sich von der Region der Stream-Gruppe unterscheidet, werden die exportierten Stream-Sitzungsdateien zwischen Regionen verschoben.

Funktionsweise

Sie müssen diesen Vorgang in einer aktiven Stream-Sitzung manuell aufrufen, um die während dieser Sitzung generierten Dateien zu exportieren. Die Stream-Sitzung muss aktiv sein, insbesondere einen der folgenden Status haben: ACTIVE CONNECTED PENDING_CLIENT_RECONNECTION, und RECONNECTING. Am Ende der Sitzung exportiert Amazon GameLift Streams die Dateien in Ihren Bucket in Amazon Simple Storage Service (Amazon S3). Somit befinden sich alle exportierten Daten in Ihrem Eigentum und unterliegen den Berechtigungsrichtlinien des Amazon S3 S3-Buckets.

Hier finden Sie eine exemplarische Vorgehensweise für den Lebenszyklus einer Stream-Sitzung mit aktivierten Exportdateien:

1. Amazon GameLift Streams beginnt eine Sitzung, indem es den Benutzer mit Ihrer Anwendung verbindet, die auf der Rechenressource ausgeführt wird.

2. Während Ihre Anwendung streamt, erstellt oder ändert sie Dateien im Dateisystem der Laufzeitumgebung.
3. Wenn die Sitzung endet, erhält Amazon GameLift Streams eine Kopie aller neuen oder geänderten Dateien im Dateisystem und exportiert die Dateien in Ihren Amazon S3 S3-Bucket.

Amazon GameLift Streams sammelt die folgenden generierten und geänderten Dateien. Finden Sie sie in den entsprechenden Ordnern im .zip Archiv.

- **application/**: Der Ordner, in dem Ihre Anwendung oder Ihr Spiel gespeichert ist.
- **profile/**: Der Profilordner des Benutzers enthält die persönlichen Einstellungen, Konfigurationen und Daten des Benutzers.
- **temp/**: Der temporäre Ordner des Systems enthält temporäre Dateien und Daten, die Ihre Anwendung und das System erstellen. Dies kann Cache-Dateien, Protokolldateien oder Zwischenverarbeitungsdaten umfassen.
- **stats/**: Enthält `perf_stats_v1.csv`, enthält Leistungsstatistiken für die Sitzung, die pro Sekunde gesammelt werden. Dazu gehören Statistiken auf Anwendungsebene (CPU- und Speicherauslastung) und Statistiken auf Systemebene (CPU-, Arbeitsspeicher-, GPU- und VRAM-Auslastung). Eine ausführliche Beschreibung der einzelnen in der CSV-Datei enthaltenen Statistiken finden Sie unter [the section called “Referenz zu Leistungsstatistiken”](#)

Um die Dateien zu löschen, löschen Sie das Objekt im Amazon S3 S3-Bucket.

Auswirkung auf die Kosten

Es fallen Kosten für die Speicherung der Dateien in Amazon S3 an. Eine Stream-Sitzung kann je nach Ihrer Anwendung eine große Datenmenge generieren. Beachten Sie, dass sich bei vielen Stream-Sitzungen, bei denen diese Funktion aktiviert ist, die Kosten summieren können.

Weitere Informationen finden Sie unter [Amazon S3 S3-Preise](#).

Dateien exportieren (Konsole)

Um den Export von Stream-Sitzungsdateien in der Amazon GameLift Streams-Konsole zu aktivieren

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).

2. Wählen Sie in der Navigationsleiste Sessions aus, um eine Liste der aktiven und letzten Stream-Sessions der letzten 90 Tage anzuzeigen.
3. Wählen Sie auf der Registerkarte Aktive Sitzungen eine aktive Stream-Sitzung aus.
4. Wählen Sie Dateien exportieren, um die Funktion zum Exportieren von Dateien für diese Stream-Sitzung zu aktivieren.
5. Wählen Sie im Dialogfeld Stream-Sitzungsdatei exportieren entweder Neuen S3-Bucket erstellen oder Bestehenden S3-Bucket auswählen aus. Folgen Sie den Schritten in der Konsole, um ein S3-Objekt zu erstellen oder auszuwählen, in dem die exportierten Daten gespeichert werden sollen.

 **Warning**

Wenn der Name der ZIP-Datei mit einem vorhandenen Namen im Verzeichnis übereinstimmt, wird der vorherige überschrieben.

6. Wählen Sie Bestätigen aus. Sie können die Sitzung jetzt auf der Registerkarte Exportierte Dateien finden.
7. Warten Sie, bis die Sitzung beendet ist und die Dateien exportiert sind.

Amazon GameLift Streams exportiert die Dateien, wenn sich die Sitzung im Status Beendet befindet. Wenn eine Sitzung beendet wurde, wechselt sie von der Registerkarte Aktive Sitzungen zur Registerkarte Letzte Sitzungen.

Sie können den Status des Exportvorgangs auf der Registerkarte Sitzungsexporte überprüfen. Wenn der Status Ausstehend lautet, ist die Stream-Sitzung immer noch aktiv, sodass Amazon GameLift Streams die Dateien noch nicht exportiert hat. Wenn der Status Erfolgreich lautet, können Sie die Dateien über den bereitgestellten Link von Amazon S3 herunterladen. Wenn der Status Fehlgeschlagen lautet, bewegen Sie den Mauszeiger über den Status, um den Grund für den Fehler zu sehen.

Dateien exportieren (CLI)

Voraussetzung

Sie müssen das AWS CLI mit Ihren Benutzeranmeldedaten und den von Ihnen ausgewählten konfigurieren AWS-Region. Anweisungen zur Einrichtung finden Sie unter [Laden Sie das herunter AWS CLI](#).

Um Stream-Sitzungsdateien zu exportieren, AWS CLI

AWS CLI Verwenden Sie dabei den [ExportStreamSessionFiles](#)Befehl, der an Ihren Inhalt angepasst ist.

```
aws gameliftstreams export-stream-session-files \
--identifier arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4 \
--stream-session-identifier arn:aws:gameliftstreams:us-
west-2:111122223333:streamsession/sg-1AB2C3De4/ABC123def4567
--output-uri s3://amzn-s3-demo-bucket/prefix
```

Wo

identifier

Ein [Amazon-Ressourcename \(ARN\)](#) oder eine Amazon-ID, die die Stream-Gruppenressource eindeutig identifiziert.

ARN-Beispiel: `arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/
sg-1AB2C3De4`

ID-Beispiel: `sg-1AB2C3De4`

stream-session-identifier

Ein [Amazon-Ressourcename \(ARN\)](#) oder eine Amazon-ID, die die Stream-Sitzungsressource eindeutig identifiziert.

ARN-Beispiel: `arn:aws:gameliftstreams:us-west-2:111122223333:streamsession/
sg-1AB2C3De4/ABC123def4567`

ID-Beispiel: `ABC123def4567`

output-uri

Die Amazon S3 S3-Bucket-URI, in die Amazon GameLift Streams den Satz komprimierter exportierter Dateien für diese Stream-Sitzung hochlädt.

Es gibt zwei gültige Formate, die Sie bereitstellen können. Wenn der URI die `.ZIP` Dateierweiterung `.zip` OR hat, speichert Amazon GameLift Streams die exportierten Dateien unter dem angegebenen URI. Andernfalls generiert Amazon GameLift Streams den Namen für

einen komprimierten Ordner und speichert ihn unter der URI. Der generierte Name folgt dem Muster:`date-time-applicationId-streamGroupId-streamSessionId`. Beispiel:

- Wenn Sie einen URI namens `angebens3://amzn-s3-demo-bucket/MyGame_Session1.zip`, speichert Amazon GameLift Streams die Dateien genau in diesem ZIP-Ordner.
- Wenn Sie einen URI namens `angebens3://amzn-s3-demo-bucket/MyGame_Session1/YYYYMMDD-HHMMSS-applicationId-streamGroupId-sessionId.zip`.

Stellen Sie sicher, dass Ihr ZIP-Dateiname den [Richtlinien zur Benennung von Objektschlüsseln](#) im Amazon Simple Storage Service-Benutzerhandbuch entspricht.

 **Warning**

Wenn der Name der ZIP-Datei mit einem vorhandenen Namen im Verzeichnis übereinstimmt, wird der vorherige überschrieben.

Sie können den Status der aktiven Sitzung überprüfen, indem Sie die [GetStreamSessionAPI](#) aufrufen. In der Zusammenfassung der Stream-Sitzung können Sie Details zum Status der exportierten Dateien abrufen. Wenn der Status Ausstehend lautet, ist die Stream-Sitzung immer noch aktiv, sodass Amazon GameLift Streams die Dateien noch nicht exportiert hat. Wenn der Status Erfolgreich lautet, navigieren Sie zur Ausgabe-URI, um die Dateien in Amazon S3 zu sehen. Wenn der Status Fehlgeschlagen lautet, überprüfen Sie das StatusReason in derExportFilesMetaData.

Backend-Service und WebClient von Amazon GameLift Streams

Mit Amazon GameLift Streams können Sie Anwendungen über einen Webbrowser streamen. Mit dem Amazon GameLift Streams Web SDK können Sie einen Backend-Streaming-Dienst einrichten. Anschließend stellen Endbenutzer über einen WebClient eine Verbindung zu einem Stream her. Sie können Ihr Spiel spielen oder über die gesamte Cloud mit Ihrer Anwendung interagieren.

Das Amazon GameLift Streams Web SDK umfasst einen Beispiel-Backend-Server und einen Beispiel-Webclient, mit denen Sie mit der Erstellung eines Backend-Service beginnen können. Sie können diese Beispiele auch verwenden, um zu testen, wie Amazon GameLift Streams streamt, ohne zusätzliche Entwicklung. Informationen zu den ersten Schritten finden Sie unter [Einrichtung eines Webservers und Clients mit Amazon GameLift Streams](#).

Themen

- [Unterstützte Browser und Eingaben](#)
- [Erforderliche Anschlüsse](#)
- [Einrichtung eines Webservers und Clients mit Amazon GameLift Streams](#)
- [Passen Sie das Erscheinungsbild des Streams an](#)
- [Präferenz für das Gebietsschema](#)
- [Umgang mit Mausbewegungen](#)
- [Datenkanalkommunikation zwischen einer Anwendung und einem WebClient](#)

Unterstützte Browser und Eingaben

Im Folgenden sind die unterstützten Plattformen und Browser für die Anzeige von Amazon GameLift Streams-Streams sowie deren kompatible Eingabe-Peripheriegeräte aufgeführt. Browser müssen außerdem mit Advanced Video Coding (AVC), auch bekannt als H.264, kompatibel sein.

Insgesamt empfehlen wir Google Chrome, Microsoft Edge oder eine benutzerdefinierte Chromium-basierte Desktop-Anwendung für das beste Endbenutzererlebnis und maximale Kompatibilität, insbesondere mit Gamecontrollern.

[Weitere Informationen darüber, welche Controller mit welchen Browsern kompatibel sind, findest du in der Web Gamepad API](#). Auch wenn einige Hinweise möglicherweise nicht für Amazon GameLift

Streams gelten, gehen wir davon aus, dass die meisten Gamecontroller erfolgreich eine Verbindung über Bluetooth herstellen können.

Betriebssystem	Browser	Input
Windows	Chrome, Edge	Tastatur, Maus, Mikrofon, Gamecontroller (einschließlich haptischem Feedback)
	Firefox	Tastatur, Maus, Mikrofon, Gamecontroller
Mac	Chrome, Edge, Safari	Tastatur, Maus, Mikrofon, Gamecontroller (im Bluetooth-Modus) (einschließlich haptischem Feedback)
	Firefox	Tastatur, Maus, Mikrofon
Linux	Chrome, Edge, Firefox	Tastatur, Maus
Android	Chrome, Edge	Einfache touch-to-mouse Emulation, Mikrofon, externe physische Maus, Tastatur und Gamecontroller (im Bluetooth-Modus)
iOS	Chrome, Edge, Firefox, Safari	Einfache touch-to-mouse Emulation, Mikrofon, externe

Betriebssystem	Browser	Input
		physische Maus, Tastatur und Gamecontroller (im Bluetooth-Modus)

Bekannte Probleme

Im Folgenden sind bekannte Probleme mit Browern und Eingaben aufgeführt:

- Safari beendet den Vollbildmodus sofort, wenn die Taste gedrückt Esc wird. Dies kann nicht außer Kraft gesetzt werden.
- „Eingebettete“ oder „In-App“ -Browseransichten LinkedIn, wie sie in mobilen Apps wie Yelp, Instagram und anderen enthalten sind, werden auf iOS nicht unterstützt. Diese neigen dazu, die WebRTC-Unterstützung des Browsers zu deaktivieren, die für interaktives Echtzeit-Streaming erforderlich ist. Wir empfehlen, nicht standardmäßige Browserzeichenfolgen zu erkennen und den Benutzer zum Öffnen in Safari aufzufordern.
- Wenn die Bildschirmauflösung in Ihrer Anwendung nicht auf 1080p eingestellt ist, kann das Maus-Tracking beeinträchtigt werden. Wir empfehlen, die Auswahl einer anderen Auflösung nach Möglichkeit zu deaktivieren. Wir empfehlen außerdem, den Fenstermodus zu deaktivieren und nur im Vollbildmodus auszuführen.
- Um Plug & Play von Gamecontrollern auf Proton zu unterstützen, obwohl sie in nativen Linux-Anwendungen nicht unterstützt werden, zeigen Spiele, die in Proton-Laufzeitumgebungen laufen, immer einen verbundenen Gamecontroller an, auch wenn keiner am Client angeschlossen ist. Dies könnte ein Problem bei Spielen sein, die zur Eingabe des Controllers auffordern, selbst wenn der Controller inaktiv und unbenutzt ist. Wir empfehlen, dass in Spielen die Eingabe-Benutzeroberfläche auf der Grundlage der letzten Eingabemethode angezeigt wird.

Einschränkungen

- Die meisten Laufzeitumgebungen unterstützen Gamecontroller, mit Ausnahme von Ubuntu 22.04 LTS. Wenn du Gamecontroller-Unterstützung benötigst, erwäge, das Spiel mit einer anderen Laufzeitumgebung zu erstellen. Eine Liste anderer Laufzeitumgebungen finden Sie unter [Laufzeitumgebungen](#).

- Die Gamecontroller PlayStation 5 und Luna werden in Firefox nicht unterstützt.
- Unterstützung für haptisches Feedback:
 - Haptisches Feedback auf den S/X Controllern der PlayStation 4- und Xbox-Serie wird in Chrome, Edge und Safari unterstützt.
 - Haptik auf dem PlayStation DualSense 5-Controller wird nur im Safari-Browser unterstützt.
 - Firefox unterstützt auf keinem Controller haptisches Feedback.
 - Android- und iOS-Geräte unterstützen auf keinem Controller haptisches Feedback.
- Die Funktion Stream testen in der Amazon GameLift Streams-Konsole unterstützt keine Mikrofone.

IPv6 Unterstützung

Streaming an IPv6 reine Clients wird nur mit Windows-Runtime-Anwendungen unterstützt.

Laufzeit	Streamen über IPv4	Rüber streamen IPv6
Microsoft Windows Server 2022-Basis	Ja	Ja
Ubuntu 22.04 LTS	Ja	Nein
Protonen-Laufzeiten	Ja	Nein

Erforderliche Anschlüsse

Um Amazon GameLift Streams zu integrieren, stellen Sie sicher, dass in Ihrer Netzwerkinfrastruktur die erforderlichen Ports offen und zugänglich sind. Im Folgenden finden Sie eine Liste der Ports, die Sie in Ihrem Netzwerk für die Kommunikation mit Amazon GameLift Streams offen haben sollten.

Port	Protokoll	Zweck
443	(HTTPS) TCP	AWS APIs, einschließlich Amazon GameLift Streams
33435-33465	UDP	Web-RTC

Einrichtung eines Webservers und Clients mit Amazon GameLift Streams

In diesem Tutorial richten Sie eine Web-Client-Anwendung ein, die den Streaming-Dienst GameLift von Amazon Streams integriert. Anschließend verwenden Sie das Amazon GameLift Streams Web SDK, eine JavaScript Bibliothek und Beispielcode, mit dem Sie beginnen können. Der Beispielcode umfasst einen einfachen Amazon GameLift Streams-Backend-Webserver und einen einfachen Webclient. Am Ende dieses Tutorials können Sie mithilfe des Beispielcodes einen Stream starten.

Wenn Sie Amazon GameLift Streams zum ersten Mal verwenden, empfehlen wir Ihnen dringend, mit dem [Ihren ersten Stream in Amazon GameLift Streams starten](#) Tutorial zu beginnen, das Sie durch das Hochladen eines Spiels auf Amazon S3 und das Testen des Streamings von der Amazon GameLift Streams-Konsole in Ihrem Browser aus führt.

Voraussetzungen

- Ein AWS-Konto mit den richtigen Anmeldeinformationen für den programmatischen Zugriff. Weitere Informationen finden Sie unter [Amazon GameLift Streams als Entwickler einrichten](#).
- Das AWS-SDK.
- Ein von Amazon GameLift Streams unterstützter Webbrowser — siehe. [Unterstützte Browser und Eingaben](#)
- Node.js — siehe [Download-Seite von Node.js](#).

Laden Sie das Web-SDK herunter

Für dieses Tutorial müssen Sie die folgenden Materialien aus dem Bereich Ressourcen der [Produktseite Erste Schritte](#) herunterladen:

- Amazon GameLift Streams Web SDK-Bundle: Dieses Paket beinhaltet Beispielcode für einen einfachen Backend-Service und einen Webclient.
- Amazon GameLift Streams Web SDK API-Referenz: Diese API-Referenz dokumentiert Amazon GameLift Streams API-Wrapper für. JavaScript

Richten Sie Ihre Streaming-Ressourcen ein

Sie benötigen Stream-Ressourcen — eine Anwendung und eine Stream-Gruppe —, um einen Stream zu starten. Insbesondere müssen Sie über Folgendes verfügen:

- Eine Anwendung im Status Bereit.
- Eine Stream-Gruppe im Status Aktiv mit verfügbarer Stream-Kapazität.
- Für das Streaming an anderen Orten als dem primären Standort muss die Anwendung die Replikation an diesen Speicherort abgeschlossen haben.

Informationen zum Einrichten einer Anwendung und einer Stream-Gruppe mithilfe der Amazon GameLift Streams-Konsole oder der Amazon GameLift Streams-CLI finden Sie unter [Bereiten Sie eine Anwendung in Amazon GameLift Streams vor](#) und [Streaming mit einer Amazon GameLift Streams-Stream-Gruppe verwalten](#). Eine end-to-end exemplarische Vorgehensweise in der Amazon GameLift Streams-Konsole finden Sie alternativ unter [Ihren ersten Stream in Amazon GameLift Streams starten](#).

Richten Sie einen Backend-Server ein

Der Backend-Server ist für Aufgaben wie die Authentifizierung von Benutzern, die Konfiguration von Stream-Parametern und die Ausführung von Amazon GameLift Streams-API-Aufrufen im Namen von Endbenutzern verantwortlich. Lesen Sie den Beispielcode und die Amazon GameLift Streams Web SDK API-Referenz, um mehr über die Einrichtung zu erfahren. Sehen Sie sich insbesondere die Datei `server.js` im Amazon GameLift Streams Web SDK-Paket an.

 **Important**

Dieser Code ist nur ein Beispielcode für Test- und Evaluierungszwecke und sollte nicht in einer Produktionskapazität verwendet werden.

Um den Beispiel-Backend-Dienst auszuführen

1. Öffnen Sie ein Terminal oder eine Befehlszeile und navigieren Sie zu dem Ordner `AmazonGameLiftStreamsWebSDK\GameLiftStreamsSampleGamePublisherService\`.
2. Führen Sie die folgenden Befehle aus:

```
npm install  
node server.js
```

Wenn der Beispiel-Back-End-Dienst ausgeführt wird, können Endbenutzer über den WebClient eine Verbindung zu einem Stream herstellen. Testen Sie den WebClient im nächsten Schritt.

Starten Sie einen WebClient

Die Web-Client-Anwendung ist dafür verantwortlich, Amazon GameLift Streams-Streams zu empfangen und zu dekodieren, an Endbenutzer zu streamen und die Webbrowsr-Benutzeroberfläche bereitzustellen, damit Endbenutzer mit der Anwendung interagieren können. Lesen Sie den Beispielcode und die Amazon GameLift Streams Web SDK API-Referenz, um mehr darüber zu erfahren, wie Sie das JavaScript Amazon GameLift Streams Web SDK in Ihre eigene Web-Client-Anwendung integrieren können. Näheres dazu finden Sie `public/index.html` im Amazon GameLift Streams Web SDK-Paket. Sie können sich auch die Quelle der Webseite ansehen, wenn Sie einen WebClient in Ihrem Browser starten.

Note

Die Windows-Laufzeit in Amazon GameLift Streams unterstützt Stream-Sitzungen über IPv4 oder IPv6. Linux- und Proton-Laufzeitumgebungen unterstützen jedoch nur Streaming Over IPv4.

Um eine Web-Client-Anwendung zu starten

1. Öffnen Sie einen Webbrowsr und navigieren Sie zu `http://localhost:port/`. Die Portnummer wird vom Backend-Server festgelegt. Standardmäßig ist dies der HTTP-Port 8000.
2. Spielen Sie das Spiel oder verwenden Sie die Software.
 - a. Um Eingaben, z. B. Ihre Maus, anzuhängen, wählen Sie Eingabe anhängen.
 - b. Um das Spiel zu beenden, wählen Sie die Esc-Taste.
 - c. Um den Serverprozess zu beenden, wählen Sie Strg+C.

Bereinigen Sie die Streaming-Ressourcen

Warning

Einer Stream-Gruppe fallen Kosten an, wenn ihr Streaming-Kapazität zugewiesen wurde, auch wenn diese Kapazität ungenutzt ist. Um unnötige Kosten zu vermeiden, skalieren Sie Ihre Stream-Gruppen auf die erforderliche Größe. Während der Entwicklung empfehlen wir, die ständig verfügbare Kapazität und die Zielkapazität im Leerlauf in Ihren Stream-Gruppen auf Null zu skalieren, wenn sie nicht verwendet werden. Weitere Informationen finden Sie unter [Skalieren Sie Stream-Gruppen auf eine Kapazität von Null](#).

Nachdem Sie das Tutorial abgeschlossen haben und Ihre Anwendung nicht mehr streamen müssen, gehen Sie wie folgt vor, um Ihre Amazon GameLift Streams-Ressourcen zu bereinigen.

Löschen einer Stream-Gruppe

Wenn Sie eine Stream-Gruppe löschen, arbeitet Amazon GameLift Streams daran, die gesamte Stream-Kapazität freizugeben.

Um eine Stream-Gruppe mit der Amazon GameLift Streams-Konsole zu löschen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Um eine Liste Ihrer vorhandenen Stream-Gruppen anzuzeigen, wählen Sie im Navigationsbereich Stream-Gruppen aus.
3. Wählen Sie den Namen der Stream-Gruppe aus, die Sie löschen möchten.
4. Wählen Sie auf der Detailseite der Stream-Gruppe die Option Löschen aus.
5. Bestätigen Sie im Dialogfeld „Löschen“ die Löschaktion.

Amazon GameLift Streams beginnt mit der Freigabe von Rechenressourcen und dem Löschen der Stream-Gruppe. Während dieser Zeit befindet sich die Stream-Gruppe im Status Löschen. Nachdem Amazon GameLift Streams die Stream-Gruppe gelöscht hat, können Sie sie nicht mehr abrufen.

Eine Anwendung löschen

Sie können eine Anwendung nur löschen, wenn sie die folgenden Bedingungen erfüllt:

- Die Anwendung hat den Status Bereit oder Fehler.
- Die Anwendung wird aktuell in keiner Stream-Sitzung gestreamt. Sie müssen warten, bis der Client die Stream-Sitzung beendet, oder die Amazon GameLift Streams-API aufrufen [TerminateStreamSession](#), um den Stream zu beenden.

Wenn die Anwendung mit Stream-Gruppen verknüpft ist, müssen Sie die Verknüpfung mit allen zugehörigen Stream-Gruppen aufheben, bevor Sie sie löschen können. In der Konsole führt Sie ein Dialogfeld durch diesen Vorgang.

Um eine Anwendung mithilfe der Amazon GameLift Streams-Konsole zu löschen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die [Amazon GameLift Streams-Konsole](#).
2. Wählen Sie in der Navigationsleiste Anwendungen aus, um eine Liste Ihrer vorhandenen Anwendungen anzuzeigen. Wählen Sie die Anwendung aus, die Sie löschen möchten.
3. Wählen Sie auf der Detailseite der Anwendung die Option Löschen aus.
4. Bestätigen Sie im Dialogfeld Löschen die Löschaktion.

Amazon GameLift Streams beginnt mit dem Löschen der Anwendung. Während dieser Zeit befindet sich die Anwendung im Deleting Status. Nachdem Amazon GameLift Streams die Anwendung gelöscht hat, können Sie sie nicht mehr abrufen.

Passen Sie das Erscheinungsbild des Streams an

Ladebildschirm

Wenn ein Kunde einen Webbrowser öffnet, um einen Stream anzusehen, beginnt der Webclient, eine Verbindung zur Amazon GameLift Streams-Streams-Sitzung herzustellen. Während die Stream-Sitzung geladen wird, können Sie auf dem Bildschirm des Kunden einen benutzerdefinierten Hintergrund und ein Logo anzeigen.

Der Amazon GameLift Streams Web SDK-Beispielclient in der [GameLiftStreamsSampleGamePublisherService/public>LoadingScreen/loadingscreen.js](#) Datei zeigt, wie Sie ein animiertes Logo in Ihrem Frontend-Webclient implementieren können. Der Standard-Ladebildschirm besteht aus 2 Bildern: Hintergrund und

Vordergrund. Das Vordergrundbild ist in der Mitte positioniert und hat eine Pulsanimation. Die Animation wird nur abgespielt, während die Stream-Sitzung eine Verbindung herstellt.

Um einen Ladebildschirm zu aktivieren

1. Navigieren Sie im Amazon GameLift Streams Web SDK-Beispielclient zu dem GameLiftStreamsSampleGamePublisherService/public>LoadingScreen/ Ordner.
2. Fügen Sie Ihre Hintergrund- und Vordergrundbilder mit den Standardnamen hinzu, Background.png und LoadingLogo.png. Wenn Sie sie umbenennen oder ein anderes Bildformat verwenden möchten, müssen Sie den Code in aktualisierenGameLiftStreamsSampleGamePublisherService/public/loadingscreen.js.
3. (Optional) Aktualisieren Sie den JavaScript Code unter GameLiftStreamsSampleGamePublisherService/public/loadingscreen.js, um verschiedene Animationen zu implementieren.

Präferenz für das Gebietsschema

In Amazon GameLift Streams können Sie die Gebietsschemaeinstellungen pro Stream festlegen. Dies ist nützlich, wenn Ihre Anwendung standortspezifische Informationen wie Uhrzeit oder Währung vom Betriebssystem des Endbenutzers abruft.

Amazon GameLift Streams unterstützt die folgenden Sprachen:

Wert	Description
en_US	US-Englisch (Standard)
ja_jp.UTF-8	Japanisch

Um die Gebietsschemaeinstellung zu ändern

Wenn Sie [StartStreamSession](#) mit der Amazon GameLift Streams-API aufrufen, fügen Sie LANG=<*language*> zu Ihrem `AdditionalEnvironmentVariables`. Da die Gebietsschemaeinstellung für jeden Benutzer einzigartig ist, legen Sie dies auf der Ebene der Stream-Sitzung fest. Wenn Sie dies nicht festlegen, verwendet der Stream standardmäßig US-Englisch.

Example Beispiel

```
aws gameliftstreams start-stream-session \
--identifier arn:aws:gameliftstreams:us-west-2:123456789012:streamgroup/1AB2C3De4 \
--protocol WebRTC \
--signal-request "[webrtc-ice-offer json string]" \
--user-id xnshijwh \
--additional-environment-variables '{"LANG": "ja_JP.UTF-8"}'
```

Umgang mit Mausbewegungen

Die Handhabung von Mausbewegungen ist entscheidend für die Bereitstellung reaktionsschneller und intuitiver Benutzererlebnisse in gestreamten Anwendungen. Amazon GameLift Streams optimiert automatisch die Übertragung von Mauseingaben auf der Grundlage des Cursorverhaltens Ihrer Anwendung und stellt so sicher, dass sich Mausbewegungen natürlich anfühlen, unabhängig davon, ob der Cursor versteckt oder sichtbar ist. Wenn Sie wissen, wie Amazon GameLift Streams Mausereignisse verarbeitet, können Sie Anwendungen entwickeln, die nahtlos mit dem Streaming-Dienst zusammenarbeiten und die bestmögliche Benutzererfahrung bieten.

Eingabemodi für die Maus

Amazon GameLift Streams verwendet zwei unterschiedliche Modi für die Übertragung von Mausereignissen an Ihre Anwendung und wählt automatisch den geeigneten Modus basierend auf der Cursorsichtbarkeit aus:

Relativer Modus

Im relativen Modus werden Mausaktualisierungen als kleine, inkrementelle Abweichungen von der vorherigen Position übertragen. Dieser Modus ist ideal für Anwendungen, die eine präzise, kontinuierliche Verfolgung der Mausbewegungen erfordern, wie z. B. Ego-Shooter-Spiele (FPS) oder Benutzeroberflächen mit 3D-Ausrichtung. Amazon GameLift Streams verwendet den relativen Modus, wenn der Betriebssystem-Cursor versteckt oder vollständig transparent ist.

Absoluter Modus

Im absoluten Modus wird die Position des Mauszeigers als exakte Bildschirmkoordinate übertragen. Dieser Modus eignet sich gut für Anwendungen, die auf eine präzise Cursorpositionierung angewiesen sind, wie z. B. point-and-click Spiele oder Benutzeroberflächen mit anklickbaren Elementen. Amazon GameLift Streams verwendet den absoluten Modus, wenn

der Betriebssystem-Cursor sichtbar ist, auch wenn Ihre Anwendung ein benutzerdefiniertes Cursorbild anzeigt.

Diese automatische Auswahl gewährleistet eine optimale Leistung für verschiedene Anwendungstypen, ohne dass eine manuelle Konfiguration erforderlich ist.

Zeigersperre

Die Zeigersperre ist eine Web-API-Funktion, die den Mauszeiger innerhalb eines bestimmten Elements erfasst, den Cursor versteckt und verhindert, dass er den dafür vorgesehenen Bereich verlässt. Diese Funktion ist besonders nützlich für Spiele, bei denen die Maus zur Kamerasteuerung oder zum Zielen uneingeschränkt bewegt werden muss, ohne dass ein sichtbarer Cursor abgelenkt wird oder Fensterränder nur eingeschränkt erreicht werden können.

Amazon GameLift Streams bietet automatische Funktionen zum Sperren von Zeigern über die `autoPointerLock` Eigenschaft in der `InputConfiguration` Oberfläche des Web-SDK. Diese Funktion ist in die [requestPointerLock API](#) integriert, um eine intuitive und kontextsensitive Mauserfassung zu ermöglichen.

Automatisches Verhalten der Zeigersperre

Amazon GameLift Streams aktiviert automatisch die Zeigersperre, wenn sich die Anwendung im Vollbildmodus befindet und der Remote-Cursor auf dem Stream-Host unsichtbar ist. Dieses Verhalten passt gut zu gängigen Mustern bei der Spieleentwicklung:

- FPS/TPS-Spiele und 3D-Orientierungssteuerung — Der Zeiger wird automatisch gesperrt und der Cursor wird ausgeblendet, sodass eine uneingeschränkte Kamerasteuerung gewährleistet ist, die für das FPS-Gameplay unerlässlich ist.
- Point-and-click Spiele und UI-Steuerung — Wenn in Spielen der Cursor für Menüinteraktionen oder Strategiespiele sichtbar ist, bleibt der Zeiger sichtbar und entsperrt, sodass die gewünschte Benutzererfahrung erhalten bleibt.

Konfigurationsoptionen

Die `autoPointerLock` Eigenschaft akzeptiert die folgenden Werte:

`true`

Die Maus wird immer erfasst, wenn der Remote-Cursor unsichtbar ist.

false

Die Maus wird nie erfasst, unabhängig von der Sichtbarkeit des Cursors.

'fullscreen' (Standard)

Die Maus wird nur aufgenommen, wenn sich das Videoelement im Vollbildmodus befindet und der Remote-Cursor unsichtbar ist.

 Important

`autoPointerLock` hat aufgrund von Plattformbeschränkungen keine Auswirkungen auf den Safari-Browser oder auf iOS-Plattformen.

Bewährte Methoden

Um eine optimale Maussteuerung in Ihren gestreamten Anwendungen zu gewährleisten:

- Streamen Sie immer im Vollbildmodus — Ihre Anwendung sollte bereits im Vollbildmodus laufen, damit sie in unserem Service ordnungsgemäß funktioniert. Darüber hinaus empfehlen wir, die Browserunterstützung zu verwenden, um den Stream zu einem Vollbild-Element zu machen, um ein optimales Endbenutzererlebnis zu erzielen. Auf diese Weise können Probleme wie Ausrichtungsprobleme zwischen dem Systemcursor und dem Software-Cursor vermieden werden.
- Den Cursor bei Relativbewegungen ausblenden — Wenn Ihre Anwendung relative Mausbewegungen erwartet (z. B. bei Kamerasteuerungen im FPS-Stil oder Interaktionen auf Ziehbasis), blenden Sie den Betriebssystem-Cursor während dieser Interaktionen aus. In einigen Szenarien müssen Sie den Cursor bei gedrückter Maus möglicherweise ausblenden und bei gedrückter Maus wieder einblenden.
- Den Cursor für absolute Positionierung anzeigen — Wenn Ihre Anwendung eine präzise Cursorpositionierung für Benutzeroberflächeninteraktionen benötigt, stellen Sie sicher, dass der Betriebssystem-Cursor sichtbar bleibt, um den absoluten Koordinatenmodus zu aktivieren.
- Testen Sie verschiedene Eingabeszenarien — Stellen Sie sicher, dass Ihre Anwendung sowohl den relativen als auch den absoluten Mausmodus korrekt verarbeitet, da Amazon GameLift Streams je nach Änderung der Cursorsichtbarkeit zwischen den Modi wechseln kann.
- Testen Sie verschiedene Fenstermodi — Testen Sie gegebenenfalls die Maussteuerung Ihrer Anwendung sowohl im Fenster- als auch im Vollbildmodus. Ermitteln Sie, welche `autoPointerLock` Einstellung für Ihre Eingabekonfiguration am besten geeignet ist.

Datenkanalkommunikation zwischen einer Anwendung und einem Webclient

Datenkanäle ermöglichen Ihnen die sichere Kommunikation beliebiger Nachrichten zwischen Ihrer Amazon GameLift Streams-Anwendung und dem Webclient (dem JavaScript Code, der im Webbrowser des Endbenutzers ausgeführt wird). Auf diese Weise können Endbenutzer über den Webbrowser, in dem sie den Stream ansehen, mit der Anwendung interagieren, die Amazon GameLift Streams streamt.

Hier sind einige Beispiele für Anwendungsfälle von Datenkanälen in Amazon GameLift Streams:

- Benutzer können die Anwendung URLs in ihrem lokalen Browser öffnen.
- Benutzer können Inhalte in der Zwischenablage hin und her an die Anwendung übergeben.
- Benutzer können Inhalte von ihrem lokalen Computer in die Anwendung hochladen.
- Entwickler können die Benutzeroberfläche im Browser implementieren, der Befehle an die Anwendung sendet.
- Benutzer können Schemas übergeben, um die Anzeige von Visualisierungsebenen zu steuern.

Features

Größenbeschränkungen für Nachrichten

Das Amazon GameLift Streams Web SDK legt eine maximale Größenbeschränkung von 64 KB (65536 Byte) pro Nachricht fest. Dadurch wird sichergestellt, dass die Größenbeschränkungen für Nachrichten mit den meisten Browzern kompatibel sind und dass die Kommunikation nur geringe Auswirkungen auf die Gesamtbandbreite des Streams hat.

Metriken

Metriken zu Ihrer Datenkanalnutzung werden an Ihr AWS-Konto gesendet, wenn eine Stream-Sitzung endet. Weitere Informationen finden Sie [Datenkanäle](#) im Abschnitt Amazon GameLift Streams überwachen.

Datenkanäle verwenden

Das Amazon GameLift Streams Web SDK bietet die `sendApplicationMessage` Funktion, die eine Nachricht als Byte-Array an die Anwendung sendet. Die Nachricht wird von einer Callback-Funktion verarbeitet, `clientConnection.applicationMessage` die Sie definieren.

Wenn der Client Nachrichten sendet, bevor die Anwendung eine Verbindung zum Datenkanalport herstellt, werden die Nachrichten in die Warteschlange gestellt. Wenn die Anwendung dann eine Verbindung herstellt, empfängt sie die Nachrichten. Wenn die Anwendung jedoch Nachrichten sendet, bevor der Client eine Verbindung zum Datenkanalport herstellt, gehen die Nachrichten verloren. Die Anwendung muss den Verbindungsstatus der Clients überprüfen, bevor sie eine Nachricht sendet.

Auf der Client-Seite

Schreiben Sie den folgenden Code in Ihre WebClient-Anwendung.

1. Definieren Sie die Callback-Funktion, um eingehende Nachrichten von der Anwendung zu empfangen.

```
function streamApplicationMessageCallback(message) {  
    console.log('Received ' + message.length + ' bytes of message from  
    Application');  
}
```

2. Stellen Sie `clientConnection.applicationMessage` Ihre Callback-Funktion ein.

```
clientConnection: {  
    connectionState: streamConnectionStateCallback,  
    channelError: streamChannelErrorCallback,  
    serverDisconnect: streamServerDisconnectCallback,  
    applicationMessage: streamApplicationMessageCallback,  
}
```

3. Rufen Sie die `GameLiftStreams.sendApplicationMessage` Funktion auf, um Nachrichten an Ihre Anwendung zu senden. Sie können dies jederzeit aufrufen, solange die Stream-Sitzung aktiv ist und die Eingabe angehängt ist.

Sehen Sie sich als Beispiel den Amazon GameLift Streams Web SDK-Beispielclient an, der zeigt, wie Sie einen einfachen Datenkanal auf der Client-Seite einrichten.

Auf der Anwendungsseite

Schreiben Sie die folgende Logik in Ihre Anwendung.

Schritt 1. Connect zum Datenkanalport her

Wenn Ihre Anwendung gestartet wird, stellen Sie eine Verbindung zum Port 40712 on herlocalhost. Ihre Anwendung sollte diese Verbindung während der gesamten Ausführungszeit aufrechterhalten. Wenn die Anwendung die Verbindung schließt, kann sie nicht erneut geöffnet werden.

Schritt 2. Achte auf Ereignisse

Ein Ereignis beginnt mit einem Header fester Größe, gefolgt von zugehörigen Daten variabler Länge. Wenn Ihre Anwendung ein Ereignis empfängt, analysieren Sie das Ereignis, um die Informationen abzurufen.

Ereignisformat

- Header: Ein 4-Byte-Header im Formular abcc
 - a: Client-ID-Byte. Dies identifiziert eine bestimmte Client-Verbindung im Fall mehrerer Verbindungen (aufgrund von Verbindungsabbruch und Wiederverbindung).
 - b: Ereignistyp Byte. 0 - der Client hat eine Verbindung hergestellt, 1 - der Client hat die Verbindung getrennt, 2 - eine Nachricht wurde vom Client gesendet. Andere Ereignistypen werden möglicherweise mit future Amazon GameLift Streams-Serviceupdates empfangen und sollten ignoriert werden.
 - cc: Länge der zugehörigen Ereignisdaten. Dies wird als 2 Byte mit Big-Endian-Reihenfolge dargestellt (das erste Byte ist das signifikanteste). Wenn der Ereignistyp 2 ist, stellen die Ereignisdaten den Nachrichteninhalt des Clients dar.
- Daten: Die verbleibenden Byte enthalten die Ereignisdaten, z. B. eine Client-Nachricht. Die Länge der Daten wird cc im Header durch angegeben.

Um auf Ereignisse zu warten

1. Lesen Sie die vier Header-Bytes, um die Client-ID, den Ereignistyp und die Länge der Ereignisdaten abzurufen.
2. Lesen Sie die Ereignisdaten variabler Länge unabhängig von der Client-ID und dem Ereignistyp entsprechend der im Header beschriebenen Länge. Es ist wichtig, die Daten bedingungslos zu lesen, damit die Ereignisdaten niemals im Puffer verbleiben, wo sie mit dem nächsten Event-Header verwechselt werden könnten. Machen Sie keine Annahmen über die Länge der Daten, die auf Ereignistypen basieren.

3. Ergreifen Sie je nach Ereignistyp die entsprechenden Maßnahmen, sofern diese von Ihrer Anwendung erkannt werden. Diese Aktion kann das Protokollieren einer eingehenden Verbindung oder Verbindungsunterbrechung oder das Analysieren der Client-Nachricht und das Auslösen der Anwendungslogik umfassen.

Schritt 3. Nachrichten an den Client übertragen

Die Anwendung sollte Nachrichten mit demselben Vier-Byte-Header-Format übertragen, das von eingehenden Ereignissen verwendet wird.

Um eine Nachricht an den Client zu übertragen

1. Schreiben Sie den Header mit den folgenden Eigenschaften:
 - a. a: Client-ID-Byte. Wenn Ihre Nachricht als Antwort auf eine Client-Nachricht gesendet wird, sollte sie dieselbe Client-ID wie die eingehende Client-Nachricht wiederverwenden, um zu verhindern, dass Racebedingungen auftreten, wie z. B. die Zustellung einer Antwort von einer alten Client-Verbindung an einen neu verbundenen Client. Wenn Ihre Anwendung eine unaufgeforderte Nachricht an den Client sendet, sollte sie die Client-ID so einstellen, dass sie dem letzten Ereignis „Client-Verbindung“ entspricht (Ereignistyp 0).
 - b. b: Der Ereignistyp ausgehender Nachrichten muss immer 2 sein. Der Client ignoriert Nachrichten mit anderen Ereignistypen.
 - c. cc: Länge der Nachricht in Byte.
2. Schreiben Sie die Nachrichten-Bytes.

Die Nachricht wird an den angegebenen Client zugestellt, sofern der Client die Verbindung nicht unterbricht. Wenn ein Client, dessen Verbindung unterbrochen wurde, erneut eine Verbindung herstellt, wird über ein Ereignis, bei dem die Verbindung zum Client hergestellt wurde, eine neue Client-ID zugewiesen. Alle nicht zugestellten Nachrichten für die alte Client-ID werden verworfen.

Example

Der folgende Pseudocode demonstriert die Logik für die Kommunikation von Nachrichten auf der Anwendungsseite. Ein vollständiges Beispiel für die Verwendung von Winsock finden Sie unter [Complete Winsock Client Code in der Windows Sockets 2-Dokumentation](#).

```
connection = connect_to_tcp_socket("localhost:40712")
loop:
    while has_pending_bytes(connection):
        client_id = read_unsigned_byte(connection)
        event_type = read_unsigned_byte(connection)
        event_length = 256 * read_unsigned_byte(connection)
        event_length = event_length + read_unsigned_byte(connection)
        event_data = read_raw_bytes(connection, event_length)
        if message_type == 0:
            app_process_client_connected(client_id)
        else if message_type == 1:
            app_process_client_disconnected(client_id)
        else if message_type == 2:
            app_process_client_message(client_id, event_data)
        else:
            log("ignoring unrecognized event type")
    while app_has_outgoing_messages():
        target_client_id, message_bytes = app_next_outgoing_message()
        message_length = length(message_bytes)
        write_unsigned_byte(connection, target_client_id)
        write_unsigned_byte(connection, 2)
        write_unsigned_byte(connection, message_length / 256)
        write_unsigned_byte(connection, message_length mod 256)
        write_raw_bytes(connection, message_bytes)
```

Checkliste für den Start von Amazon GameLift Streams

Die Vorbereitung eines erfolgreichen Starts bei Amazon GameLift Streams erfordert Planung und Koordination. Folgen Sie dieser detaillierten Checkliste, um einen reibungslosen Ablauf in den Wochen vor Ihrer Veranstaltung zu gewährleisten.

Das Amazon GameLift Streams-Team benachrichtigen

Maßnahme: Informieren Sie Ihren technischen Kundenbetreuer, Ihr Account-Team oder Ihren Account Solution Architect mindestens 8 Wochen im Voraus über Ihren Zeitplan für die Markteinführung und die zu erwartenden Spitzenwerte bei gleichzeitigen Streams.

Grund: Wenn wir den Umfang Ihrer Produktionsauslastung kennen, können wir sicherstellen, dass Ihre Servicebeschränkungen angemessen sind, und sie gegebenenfalls anpassen. Wir geben auch Hinweise zur Kapazitätsverfügbarkeit und geben Empfehlungen für die Markteinführung.

Kompatibilitäts- und Leistungstests

Maßnahme: Testen Sie Ihre Anwendung in großem Umfang und an allen Standorten, an denen Sie Kapazitäten haben, um ein positives Kundenerlebnis zu bestätigen. Amazon GameLift Streams bietet NVIDIA-basierte Stream-Klassen mit unterschiedlichen Leistungsstufen und unterstützten Laufzeiten.

Grund: Gründliche Tests helfen dabei, potenzielle Kompatibilitäts- und Leistungsprobleme vor der Markteinführung zu identifizieren und zu lösen. Beachten Sie bei Stream-Klassen Folgendes:

- Die Stream-Klassen „High“ unterstützen Mehrmandantenfähigkeit, sodass zwei Anwendungen gleichzeitig auf einer einzigen Instanz ausgeführt werden können. Wenn Sie die Stream-Klasse „High“ verwenden, testen Sie mit mindestens 2 gleichzeitigen Streams, um zu sehen, wie Ihre Anwendung mit gemeinsam genutzten Ressourcen wie CPU, GPU und Arbeitsspeicher abschneidet.

Reservierung von Kapazitäten

Maßnahme: Wenden Sie sich mindestens 8 Wochen vor der Markteinführung an Ihr Account-Team, um Kapazitäten zu reservieren, insbesondere, wenn Sie einen kritischen, umfangreichen Bedarf erwarten. Entscheiden Sie sich anhand Ihrer Kompatibilitätstests, Leistungsanforderungen und Ihres Budgets für die Streaming-Klassen und Streaming-Standorte. Geben Sie die start/end Zeiten und die

erforderliche Kapazität an. AWS erfordert, dass alle Kapazitätsreservierungen 6-8 Wochen vor dem Mindesthaltbarkeitsdatum der Reservierung abgeschlossen werden.

Grund: Amazon GameLift Streams arbeitet nach dem Prinzip „Wer zuerst kommt, mahlt zuerst“ und nutzt On-Demand-Kapazität. Reservierungen sind unerlässlich, um die erforderliche Kapazität zu gewährleisten.

Leistungstests im großen Maßstab

Maßnahme: Führen Sie gründliche Auslastungstests Ihrer APIs und Ihrer Amazon GameLift Streams-Konfigurationen durch, um deren Leistung unter Last zu beobachten (Latenz, Auflösung und Bildrate). Achten Sie darauf, das zu überprüfen, [Amazon GameLift Streams API-Ratenbegrenzungen](#) um sicherzustellen, dass Sie genügend Spielraum für Ihre Markteinführung und darüber hinaus haben. Wenn Sie der Meinung sind, dass Sie eine Erhöhung des Limits benötigen, sollten Sie sich an Ihren Account Manager wenden oder ein Support-Ticket einreichen.

Grund: Lasttests zeigen, wie sich Ihre Anwendung und Amazon GameLift Streams-Konfigurationen vor dem Start unter Stress verhalten. Dies ist entscheidend, um eine reibungslose Leistung im großen Maßstab zu gewährleisten.

Einrichtung vor dem Start

Aktion: Erstellen Sie mindestens 2-3 Tage vor dem Start Ihre endgültigen Anwendungsressourcen und Stream-Gruppen. Überprüfen Sie die Streaming-Leistung und erhöhen Sie die Kapazität nach Bedarf.

Grund: Dadurch wird sichergestellt, dass alle Komponenten wie erwartet funktionieren, wodurch das Risiko unerwarteter Probleme minimiert wird und die Diagnose und Wiederherstellung während des Ereignisses erleichtert wird.

Zusätzliche Tipps

- Konsistenz ist entscheidend: Die Verwendung derselben vorhandenen Stream-Gruppen während eines Launch-Events gewährleistet die Konsistenz im Amazon GameLift Streams-Backend und vereinfacht so die Fehlerbehebung.
- Genaue Überwachung: Überwachen Sie die Leistung und das Feedback der Benutzer genau, um Probleme schnell zu beheben. Erstellen Sie ein betriebsbereites Dashboard. Überwachen Sie die

Kapazität, Nutzung und Leistung von Streams mithilfe von Amazon CloudWatch ([Überwachen Sie mit CloudWatch](#)weitere Informationen finden Sie unter). Weitere Hinweise finden Sie im [Well-Architected Framework](#).

Benötigen Sie weitere Unterstützung?

Wenn Sie Fragen haben oder weitere Unterstützung benötigen, wenden Sie sich bitte an den [Amazon GameLift Streams-Support](#). Wir sind hier, um sicherzustellen, dass Ihr Start erfolgreich und reibungslos verläuft.

Sicherheit in Amazon GameLift Streams

Cloud-Sicherheit AWS hat höchste Priorität. Als AWS Kunde profitieren Sie von Rechenzentren und Netzwerkarchitekturen, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame AWS Verantwortung von Ihnen und Ihnen. Das [Modell der geteilten Verantwortung](#) beschreibt dies als Sicherheit der Cloud und Sicherheit in der Cloud:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, auf der AWS Dienste in der ausgeführt AWS Cloud werden. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Externe Prüfer testen und verifizieren regelmäßig die Wirksamkeit unserer Sicherheitsmaßnahmen im Rahmen der [AWS](#). Weitere Informationen zu den Compliance-Programmen, die für Amazon GameLift Streams gelten, finden Sie unter [AWS Services im Umfang nach Compliance-Programm AWS](#).
- Sicherheit in der Cloud — Ihre Verantwortung richtet sich nach dem AWS Service, den Sie nutzen. Amazon GameLift Streams ist für die Ausführung von Programmen konzipiert, die Sie bereitstellen, und Sie sind allein für den Inhalt und die Sicherheit dieser Programme verantwortlich. Sie sind auch für andere Faktoren verantwortlich, etwa für die Vertraulichkeit Ihrer Daten, für die Anforderungen Ihres Unternehmens und für die geltenden Gesetze und Vorschriften.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Nutzung von Amazon GameLift Streams anwenden können. In den folgenden Themen erfahren Sie, wie Sie Amazon GameLift Streams konfigurieren, um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie erfahren auch, wie Sie andere AWS Dienste nutzen können, die Ihnen helfen, Ihre Amazon GameLift Streams-Ressourcen zu überwachen und zu sichern.

Topics

- [Datenschutz in Amazon GameLift Streams](#)
- [Identity and Access Management für Amazon GameLift Streams](#)
- [Konformitätsprüfung für Amazon GameLift Streams](#)
- [Resilienz in Amazon GameLift Streams](#)
- [Infrauktursicherheit in Amazon GameLift Streams](#)
- [Konfiguration und Schwachstellenanalyse in Amazon GameLift Streams](#)
- [Bewährte Sicherheitsmethoden für Amazon GameLift Streams](#)

Datenschutz in Amazon GameLift Streams

Das AWS [Modell](#) der mit gilt für den Datenschutz in Amazon GameLift Streams. Wie in diesem Modell beschrieben, AWS ist verantwortlich für den Schutz der globalen Infrastruktur, auf der alle Systeme laufen AWS Cloud. Sie sind dafür verantwortlich, die Kontrolle über Ihre in dieser Infrastruktur gehosteten Inhalte zu behalten. Sie sind auch für die Sicherheitskonfiguration und die Verwaltungsaufgaben für die von Ihnen verwendeten AWS-Services verantwortlich. Weitere Informationen zum Datenschutz finden Sie unter [Häufig gestellte Fragen zum Datenschutz](#). Informationen zum Datenschutz in Europa finden Sie im Blog-Beitrag [AWS -Modell der geteilten Verantwortung und in der DSGVO](#) im AWS -Sicherheitsblog.

Aus Datenschutzgründen empfehlen wir, dass Sie AWS-Konto Anmeldeinformationen schützen und einzelne Benutzer mit AWS IAM Identity Center oder AWS Identity and Access Management (IAM) einrichten. So erhält jeder Benutzer nur die Berechtigungen, die zum Durchführen seiner Aufgaben erforderlich sind. Außerdem empfehlen wir, die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto die Multi-Faktor-Authentifizierung (MFA).
- Wird verwendet SSL/TLS , um mit AWS Ressourcen zu kommunizieren. Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein AWS CloudTrail. Informationen zur Verwendung von CloudTrail Pfaden zur Erfassung von AWS Aktivitäten finden Sie unter [Arbeiten mit CloudTrail Pfaden](#) im AWS CloudTrail Benutzerhandbuch.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen darin enthaltenen Standardsicherheitskontrollen AWS-Services.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu schützen.
- Wenn Sie für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine API FIPS 140-3-validated kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter [Federal Information Processing Standard \(FIPS\) 140-3](#).

Wir empfehlen dringend, in Freitextfeldern, z. B. im Feld Name, keine vertraulichen oder sensiblen Informationen wie die E-Mail-Adressen Ihrer Kunden einzugeben. Dies gilt auch, wenn Sie mit Amazon GameLift Streams oder anderen AWS-Services über die Konsole AWS CLI, API oder arbeiten AWS SDKs. Alle Daten, die Sie in Tags oder Freitextfelder eingeben, die für Namen verwendet werden, können für Abrechnungs- oder Diagnoseprotokolle verwendet werden.

Wenn Sie eine URL für einen externen Server bereitstellen, empfehlen wir dringend, keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL einzuschließen.

Amazon GameLift Streams verarbeitet servicespezifische Daten wie folgt:

- Vom Kunden bereitgestellte Anwendungen — Amazon GameLift Streams speichert Kundendaten, sofern diese bereitgestellt werden, in internen, serviceverwalteten Amazon S3 S3-Buckets und auf NVME-Speicherlaufwerken, die mit Amazon-Instances verbunden sind. Alle Daten werden im Ruhezustand mit serviceverwalteter Verschlüsselung gespeichert. Es gibt keinen direkten Kundenzugriff auf diese Kopie der Daten. Um eine Anwendung zu löschen, verwenden Sie die Amazon GameLift Streams-Konsole oder die Service-API.
- Vom Kunden bereitgestellte Metadaten — Kunden können Amazon GameLift Streams Metadaten zur Verfügung stellen, APIs darunter Beschreibungen, Verbindungsinformationen und undurchsichtige Identifikatoren wie Kunden. IDs Diese Metadaten sind immer mit bestimmten Kundenressourcen verknüpft.
- Kundengenerierte Daten — Wenn eine Anwendung im Rahmen ihres normalen Betriebs neue Daten schreibt, werden diese vom Kunden generierten Daten bis zum Ende der Benutzersitzung aufbewahrt. Am Ende der Sitzung können die generierten Daten optional in ein Amazon S3 S3-Bucket-Ziel nach Wahl des Kunden exportiert werden. Kundengenerierte Daten verlassen andernfalls nicht die EC2 Amazon-Instance, in der sie generiert wurden. Weitere Informationen zur Datenverarbeitung finden Sie in den Themen zur [Sitzungsisolierung](#).
- Metriken und Ereignisdaten — Metrik- und Ereignisdaten von Amazon GameLift Streams, auf die über die Amazon GameLift Streams-Konsole oder durch Aufrufe der Service-API zugegriffen werden kann. Daten sind für Anwendungen, Stream-Gruppen und Stream-Sitzungen verfügbar. Autorisierte Benutzer können auch über Amazon CloudWatch und CloudWatch Events auf diese Daten zugreifen.

 **Important**

Wenn Sie Amazon GameLift Streams Kunden IDs - oder andere Identifikatoren zur Verfügung stellen, wird erwartet, dass es sich bei diesen Werten um anonymisierte Verweise handelt und sie keine sensiblen oder persönlichen Informationen enthalten. Amazon GameLift Streams redigiert keine Metadatenfelder.

Weitere Informationen zum Datenschutz enthält der Blog-Beitrag [AWS Shared Responsibility Model and GDPR](#) im AWS -Sicherheitsblog.

Verschlüsselung im Ruhezustand

Die Verschlüsselung von Amazon GameLift Streams-spezifischen Daten im Ruhezustand wird wie folgt gehandhabt:

- Anwendungsinhalte werden in serviceverwalteten verschlüsselten Amazon S3 S3-Buckets und zusätzlich auf hardwareverschlüsselten NVME-Laufwerken gespeichert, die an serviceverwaltete Amazon-Instances angeschlossen sind. EC2

Verschlüsselung während der Übertragung

Aufrufe der Amazon GameLift Streams APIs werden über eine sichere (SSL) Verbindung getätigt und mit [AWS Signature Version 4](#) authentifiziert (bei einer Verbindung über die AWS CLI oder das AWS SDK erfolgt die Signierung automatisch). Aufrufende Entitäten verwenden Sicherheitsanmeldedaten, die durch Anwendung der IAM-Zugriffsrichtlinien authentifiziert werden, die für Amazon GameLift Streams-Ressourcen definiert sind.

Im Kontext von Stream-Gruppen mit mehreren Standorten repliziert Amazon GameLift Streams Anwendungen sicher an diese Standorte, um eine Anwendung von einem beliebigen Standort in der Stream-Gruppe zu streamen, dem Streaming-Kapazität zugewiesen wurde.

In ähnlicher Weise speichert Amazon GameLift Streams am Ende einer Sitzung Protokolldaten und Sitzungsdateien auf Anfrage in vom Kunden benannten Amazon S3 S3-Buckets. Wenn sich der Bucket nicht am selben Ort wie die Sitzung befindet, überträgt Amazon GameLift Streams die Dateien sicher an den Ort, an AWS-Region dem sich der Bucket befindet.

Schutz von Endbenutzer-Streams

Einzelne Endbenutzer-Streams sind direkte Verbindungen zwischen dem Webbrower des Endnutzers und den Amazon GameLift Streams-Backend-Hosts. Diese Streams sind mit der branchenüblichen WebRTC-Verschlüsselung geschützt, und beide Endpunkte des Streams werden durch kryptografische Identifikatoren eindeutig identifiziert, die Teil der SignalRequest und der SignalResponse Werte sind, die während der Stream-Sitzung ausgehandelt werden. APIs

Datenkanalnachrichten werden auch durch die für Streams verwendete WebRTC-Verschlüsselung abgedeckt. Diese Nachrichten werden von Amazon GameLift Streams entschlüsselt und lokal über

eine unverschlüsselte API on-the-host an die Anwendung des Kunden weitergegeben. Wenn eine end-to-end Verschlüsselung erforderlich ist, die selbst der Amazon GameLift Streams-Service nicht entschlüsseln kann, liegt diese zusätzliche Verschlüsselungsebene in der Verantwortung des Anwendungsentwicklers.

Sitzungsisolierung in Linux-Stream-Klassen

In Linux-Stream-Klassen (Ubuntu- und Proton-Laufzeiten) verwendet Amazon GameLift Streams die Container-Isolierung. Jede Sitzung läuft in einem neuen Linux-Container, der nach der Verwendung verworfen wird. Das bedeutet, dass jede neue Sitzung in einer neuen Umgebung ausgeführt wird, isoliert von anderen Benutzern, die sich die Rechenressource teilen (wenn sie in einer Stream-Klasse mit gemeinsam genutzten Ressourcen ausgeführt wird). Wenn eine neue Sitzung gestartet wird, sind keine Daten aus früheren Sitzungen vorhanden.

Sitzungsisolierung in Windows-Stream-Klassen

Bei Windows-Stream-Klassen (Microsoft Windows Server-Laufzeiten) verwendet Amazon GameLift Streams Softwareisolierung. Der Service stützt sich auf einen Softwareagenten, der den kritischen Systemstatus zwischen den Sitzungen zurücksetzt. Einige Ordner werden über mehrere Sitzungen hinweg beibehalten, um Leistungsoptimierungen zu ermöglichen, wie z. B. das Zwischenspeichern von Festplatten auf dem Host. Der Software-Agent entfernt automatisch alle Dateien, die während der vorherigen Stream-Sitzung im Profilverzeichnis des Benutzers generiert wurden. Der Agent entfernt jedoch keine Dateien, die vor der Ausführung der Anwendung existierten und während der Ausführung der Anwendung geändert wurden. Es werden auch keine Windows-Registrierungsschlüssel entfernt, die von der Anwendung hinzugefügt wurden. Kunden sollten sich bewusst sein, dass es in ihrer Verantwortung liegt, die Integrität des gesamten Betriebssystems nicht zu beeinträchtigen. Anwendungen werden als Administratorbenutzer ausgeführt, was Änderungen an wichtigen Dateien auf Systemebene ermöglichen kann, einschließlich Änderungen, die über mehrere Sitzungen hinweg bestehen bleiben. Es liegt in der Verantwortung des Kunden, seine Anwendungen zu sichern und vor unsicheren oder instabilen Betriebssystemänderungen zu schützen.

Die Kunden sind dafür verantwortlich, die geänderten Dateien und hinzugefügten Registrierungsschlüssel aus früheren Sitzungen zu bereinigen, wenn die Anwendung gestartet wird. Dies ist ein wichtiger Schritt zum Schutz vertraulicher oder sensibler Informationen, die die Anwendung in das Profilverzeichnis des Benutzers schreibt. Zu diesem Zweck können Kunden ihr eigenes benutzerdefiniertes Skript schreiben, das die folgenden Aktionen ausführt:

- Stellen Sie alle Dateien außerhalb des %USERPROFILE% Verzeichnisses wieder her, die von der Anwendung geändert wurden.
- Bereinigen Sie alle vertraulichen oder benutzerspezifischen Registrierungsschlüssel, die die Anwendung hinzugefügt hat.

Verwaltung von Verschlüsselungsschlüsseln

Der Dienst verwendet AWS-verwaltete Verschlüsselungsschlüssel. Jede Region verwendet einen separaten KMS-Schlüssel. Vom Kunden verwaltete Schlüssel (CMKs) werden nicht unterstützt.

Für Amazon GameLift Streams bereitgestellte Anwendungsdateien können nicht erneut veröffentlicht oder aus dem Service exportiert werden. Der Kunde kann die Servicekonsole verwenden oder APIs Anwendungen löschen. Laufwerke, auf denen sich zuvor diese Anwendungsdateien befanden, können vollständig gelöscht werden, indem die zugehörigen Stream-Gruppen gelöscht werden.

Datenschutz für den Datenverkehr zwischen Netzwerken

Amazon GameLift Streams verwendet öffentlich zugängliche Netzwerke, um Stream-Sitzungen zu hosten. Jede Stream-Gruppe besteht aus einem oder mehreren service-verwalteten VPC-Netzwerken, die von anderen Stream-Gruppen und von anderen Kunden isoliert sind. Eingehende Netzwerkverbindungen werden verweigert, mit Ausnahme von authentifizierten, vom Service vermittelten WebRTC-Stream-Verbindungen. Kundenanwendungen können von diesen VPCs aus ohne Einschränkungen Verbindungen zu anderen öffentlichen Adressen herstellen.

Darüber hinaus gibt es für einen Kunden keine Möglichkeit, einen Stream oder seine Anwendungsdaten allein mithilfe von API-Aufrufen oder Einstellungen des Dienstes öffentlich zugänglich zu machen. Alle Dienstinteraktionen werden durch AWS-authentifizierte API-Aufrufe gesteuert. Wenn der Kunde einen Stream der Öffentlichkeit zugänglich machen möchte, muss er seine eigene Client-Webanwendung erstellen, die die authentifizierten Aufrufe durchführt, um einen Stream zu starten und anzuzeigen.

Identity and Access Management für Amazon GameLift Streams

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service , den Zugriff auf Ressourcen sicher zu AWS kontrollieren. IAM-Administratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um Amazon GameLift Streams-Ressourcen zu verwenden. IAM ist ein Programm AWS-Service , das Sie ohne zusätzliche Kosten nutzen können.

Themen

- [Zielgruppe](#)
- [Authentifizierung mit Identitäten](#)
- [Verwalten des Zugriffs mit Richtlinien](#)
- [So funktioniert Amazon GameLift Streams mit IAM](#)
- [Beispiele für identitätsbasierte Richtlinien für Amazon Streams GameLift](#)
- [Fehlerbehebung bei Identität und Zugriff auf Amazon GameLift Streams](#)

Zielgruppe

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von Ihrer Rolle ab:

- Servicebenutzer – Fordern Sie von Ihrem Administrator Berechtigungen an, wenn Sie nicht auf Features zugreifen können (siehe [Fehlerbehebung bei Identität und Zugriff auf Amazon GameLift Streams](#)).
- Serviceadministrator – Bestimmen Sie den Benutzerzugriff und stellen Sie Berechtigungsanfragen (siehe [So funktioniert Amazon GameLift Streams mit IAM](#)).
- IAM-Administrator – Schreiben Sie Richtlinien zur Zugriffsverwaltung (siehe [Beispiele für identitätsbasierte Richtlinien für Amazon Streams GameLift](#)).

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen sich als IAM-Benutzer authentifizieren oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich als föderierte Identität anmelden, indem Sie Anmeldeinformationen aus einer Identitätsquelle wie AWS IAM Identity Center (IAM Identity Center), Single Sign-On-Authentifizierung oder Anmeldeinformationen verwenden. Google/Facebook Weitere Informationen zum Anmelden finden Sie unter [So melden Sie sich bei Ihrem AWS-Konto an](#) im Benutzerhandbuch für AWS-Anmeldung .

AWS bietet für den programmatischen Zugriff ein SDK und eine CLI zum kryptografischen Signieren von Anfragen. Weitere Informationen finden Sie unter [AWS Signature Version 4 for API requests](#) im IAM-Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, dem sogenannten AWS-Konto Root-Benutzer, der vollständigen Zugriff auf alle AWS-Services Ressourcen hat. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Eine Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter [Tasks that require root user credentials](#) im IAM-Benutzerhandbuch.

Verbundidentität

Als bewährte Methode sollten menschliche Benutzer für den Zugriff AWS-Services mithilfe temporärer Anmeldeinformationen einen Verbund mit einem Identitätsanbieter verwenden.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensverzeichnis, Ihrem Directory Service Web-Identitätsanbieter oder der AWS-Services mithilfe von Anmeldeinformationen aus einer Identitätsquelle zugreift. Verbundene Identitäten übernehmen Rollen, die temporäre Anmeldeinformationen bereitstellen.

Für die zentrale Zugriffsverwaltung empfehlen wir AWS IAM Identity Center. Weitere Informationen finden Sie unter [Was ist IAM Identity Center?](#) im AWS IAM Identity Center -Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein [IAM-Benutzer](#) ist eine Identität mit bestimmten Berechtigungen für eine einzelne Person oder Anwendung. Wir empfehlen die Verwendung temporärer Anmeldeinformationen anstelle von IAM-Benutzern mit langfristigen Anmeldeinformationen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter [Erfordern, dass menschliche Benutzer den Verbund mit einem Identitätsanbieter verwenden müssen, um AWS mithilfe temporärer Anmeldeinformationen darauf zugreifen zu können](#).

Eine [IAM-Gruppe](#) spezifiziert eine Sammlung von IAM-Benutzern und erleichtert die Verwaltung von Berechtigungen für große Gruppen von Benutzern. Weitere Informationen finden Sie unter [Anwendungsfälle für IAM-Benutzer](#) im IAM-Benutzerhandbuch.

IAM-Rollen

Eine [IAM-Rolle](#) ist eine Identität mit spezifischen Berechtigungen, die temporäre Anmeldeinformationen bereitstellt. Sie können eine Rolle übernehmen, indem Sie [von einer Benutzer zu einer IAM-Rolle \(Konsole\) wechseln](#) AWS CLI oder einen AWS API-Vorgang aufrufen. Weitere Informationen finden Sie unter [Methoden, um eine Rolle zu übernehmen](#) im IAM-Benutzerhandbuch.

IAM-Rollen sind nützlich für Verbundbenutzerzugriff, temporäre IAM-Benutzerberechtigungen, kontoübergreifenden Zugriff, dienstübergreifenden Zugriff und Anwendungen, die auf Amazon ausgeführt werden. EC2 Weitere Informationen finden Sie unter [Kontoübergreifender Ressourcenzugriff in IAM](#) im IAM-Benutzerhandbuch.

Verwalten des Zugriffs mit Richtlinien

Sie kontrollieren den Zugriff, AWS indem Sie Richtlinien erstellen und diese an Identitäten oder Ressourcen anhängen. AWS Eine Richtlinie definiert Berechtigungen, wenn sie mit einer Identität oder Ressource verknüpft sind. AWS bewertet diese Richtlinien, wenn ein Principal eine Anfrage stellt. Die meisten Richtlinien werden AWS als JSON-Dokumente gespeichert. Weitere Informationen zu JSON-Richtliniendokumenten finden Sie unter [Übersicht über JSON-Richtlinien](#) im IAM-Benutzerhandbuch.

Mit Hilfe von Richtlinien legen Administratoren fest, wer Zugriff auf was hat, indem sie definieren, welches Prinzipal welche Aktionen auf welchen Ressourcen und unter welchen Bedingungen durchführen darf.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator erstellt IAM-Richtlinien und fügt sie zu Rollen hinzu, die die Benutzer dann übernehmen können. IAM-Richtlinien definieren Berechtigungen unabhängig von der Methode, die zur Ausführung der Operation verwendet wird.

Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität (Benutzer, Gruppe oder Rolle) anfügen können. Diese Richtlinien steuern, welche Aktionen Identitäten für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter [Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien](#) im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien können Inline-Richtlinien (direkt in eine einzelne Identität eingebettet) oder verwaltete Richtlinien (eigenständige Richtlinien, die mit mehreren Identitäten verbunden sind) sein. Informationen dazu, wie Sie zwischen verwalteten und Inline-Richtlinien wählen, finden Sie unter [Choose between managed policies and inline policies](#) im IAM-Benutzerhandbuch.

Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele hierfür sind Vertrauensrichtlinien für IAM-Rollen und Amazon S3-Bucket-

Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Sie müssen in einer ressourcenbasierten Richtlinie [einen Prinzipal angeben](#).

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

Weitere Richtlinientypen

AWS unterstützt zusätzliche Richtlinientypen, mit denen die maximalen Berechtigungen festgelegt werden können, die durch gängigere Richtlinientypen gewährt werden:

- Berechtigungsgrenzen – Eine Berechtigungsgrenze legt die maximalen Berechtigungen fest, die eine identitätsbasierte Richtlinie einer IAM-Entität erteilen kann. Weitere Informationen finden Sie unter [Berechtigungsgrenzen für IAM-Entitäten](#) im -IAM-Benutzerhandbuch.
- Richtlinien zur Dienstkontrolle (SCPs) — Geben Sie die maximalen Berechtigungen für eine Organisation oder Organisationseinheit in an AWS Organizations. Weitere Informationen finden Sie unter [Service-Kontrollrichtlinien](#) im AWS Organizations -Benutzerhandbuch.
- Richtlinien zur Ressourcenkontrolle (RCPs) — Legen Sie die maximal verfügbaren Berechtigungen für Ressourcen in Ihren Konten fest. Weitere Informationen finden Sie im AWS Organizations Benutzerhandbuch unter [Richtlinien zur Ressourcenkontrolle \(RCPs\)](#).
- Sitzungsrichtlinien – Sitzungsrichtlinien sind erweiterte Richtlinien, die als Parameter übergeben werden, wenn Sie eine temporäre Sitzung für eine Rolle oder einen Verbundbenutzer erstellen. Weitere Informationen finden Sie unter [Sitzungsrichtlinien](#) im IAM-Benutzerhandbuch.

Mehrere Richtlinientypen

Wenn für eine Anfrage mehrere Arten von Richtlinien gelten, sind die daraus resultierenden Berechtigungen schwieriger zu verstehen. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie unter [Bewertungslogik für Richtlinien](#) im IAM-Benutzerhandbuch.

So funktioniert Amazon GameLift Streams mit IAM

Bevor Sie IAM verwenden, um den Zugriff auf Amazon GameLift Streams zu verwalten, sollten Sie sich darüber informieren, welche IAM-Funktionen für Amazon GameLift Streams verfügbar sind.

IAM-Funktionen, die Sie mit Amazon GameLift Streams verwenden können

IAM-Feature	Unterstützung für Amazon GameLift Streams
<u>Identitätsbasierte Richtlinien</u>	Ja
<u>Ressourcenbasierte Richtlinien</u>	Nein
<u>Richtlinienaktionen</u>	Ja
<u>Richtlinienressourcen</u>	Ja
<u>Richtlinienbedingungsschlüssel (servicespezifisch)</u>	Ja
<u>ACLs</u>	Nein
<u>ABAC (Tags in Richtlinien)</u>	Teilweise. ABAC wird nur für Anwendungen und Stream-Gruppen unterstützt.
<u>Temporäre Anmeldeinformationen</u>	Ja
<u>Prinzipalberechtigungen</u>	Ja
<u>Servicerollen</u>	Nein
<u>Serviceverknüpfte Rollen</u>	Nein

Einen allgemeinen Überblick darüber, wie Amazon GameLift Streams und andere AWS Dienste mit den meisten IAM-Funktionen funktionieren, finden Sie im [IAM-Benutzerhandbuch unter AWS Dienste, die mit IAM funktionieren](#).

Identitätsbasierte Richtlinien für Amazon Streams GameLift

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter

[Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien](#) im IAM-Benutzerhandbuch.

Mit identitätsbasierten IAM-Richtlinien können Sie angeben, welche Aktionen und Ressourcen zugelassen oder abgelehnt werden. Darüber hinaus können Sie die Bedingungen festlegen, unter denen Aktionen zugelassen oder abgelehnt werden. Informationen zu sämtlichen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie in der [IAM-Referenz für JSON-Richtlinienelemente](#) im IAM-Benutzerhandbuch.

Beispiele für identitätsbasierte Richtlinien für Amazon Streams GameLift

Beispiele für identitätsbasierte Richtlinien von Amazon GameLift Streams finden Sie unter. [Beispiele für identitätsbasierte Richtlinien für Amazon Streams GameLift](#)

Ressourcenbasierte Richtlinien innerhalb von Amazon Streams GameLift

Unterstützt ressourcenbasierte Richtlinien: Nein

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie [einen Prinzipal angeben](#). Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Um kontoübergreifenden Zugriff zu ermöglichen, können Sie ein gesamtes Konto oder IAM-Entitäten in einem anderen Konto als Prinzipal in einer ressourcenbasierten Richtlinie angeben. Weitere Informationen finden Sie unter [Kontoübergreifender Ressourcenzugriff in IAM](#) im IAM-Benutzerhandbuch.

Politische Maßnahmen für Amazon GameLift Streams

Unterstützt Richtlinienaktionen: Ja

Administratoren können AWS JSON-Richtlinien verwenden, um festzulegen, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element Action einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Nehmen Sie Aktionen in eine Richtlinie auf, um Berechtigungen zur Ausführung des zugehörigen Vorgangs zu erteilen.

Richtlinienaktionen in Amazon GameLift Streams verwenden das folgende Präfix vor der Aktion:

```
gameliftstreams
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Komma:

Example

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

Beispiele für identitätsbasierte Richtlinien von Amazon GameLift Streams finden Sie unter. [Beispiele für identitätsbasierte Richtlinien für Amazon Streams GameLift](#)

Richtlinienressourcen für Amazon GameLift Streams

Unterstützt Richtlinienressourcen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das JSON-Richtlinienelement Resource gibt die Objekte an, auf welche die Aktion angewendet wird. Als Best Practice geben Sie eine Ressource mit dem zugehörigen [Amazon-Ressourcennamen \(ARN\)](#) an. Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

```
"Resource": "*"
```

Beispiele für identitätsbasierte Richtlinien von Amazon GameLift Streams finden Sie unter. [Beispiele für identitätsbasierte Richtlinien für Amazon Streams GameLift](#)

Schlüssel für Richtlinienbedingungen für Amazon GameLift Streams

Unterstützt servicespezifische Richtlinienbedingungsschlüssel: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Condition` gibt an, wann Anweisungen auf der Grundlage definierter Kriterien ausgeführt werden. Sie können bedingte Ausdrücke erstellen, die [Bedingungsoperatoren](#) verwenden, z. B. ist gleich oder kleiner als, damit die Bedingung in der Richtlinie mit Werten in der Anforderung übereinstimmt. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter [Kontextschlüssel für AWS globale Bedingungen](#) im IAM-Benutzerhandbuch.

Beispiele für identitätsbasierte Richtlinien von Amazon GameLift Streams finden Sie unter. [Beispiele für identitätsbasierte Richtlinien für Amazon Streams GameLift](#)

ACLs in Amazon GameLift Streams

Unterstützt ACLs: Nein

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

ABAC mit Amazon Streams GameLift

Unterstützt ABAC (Tags in Richtlinien): Teilweise

Die attributbasierte Zugriffskontrolle (ABAC) ist eine Autorisierungsstrategie, bei der Berechtigungen basierend auf Attributen, auch als Tags bezeichnet, definiert werden. Sie können Tags an IAM-Entitäten und AWS -Ressourcen anhängen und dann ABAC-Richtlinien entwerfen, um Operationen zu ermöglichen, wenn das Tag des Prinzipals mit dem Tag auf der Ressource übereinstimmt.

Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungselement einer [Richtlinie Tag-Informationen](#) an, indem Sie die Schlüssel `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, oder Bedingung `aws:TagKeys` verwenden.

Wenn ein Service alle drei Bedingungsschlüssel für jeden Ressourcentyp unterstützt, lautet der Wert für den Service Ja. Wenn ein Service alle drei Bedingungsschlüssel für nur einige Ressourcentypen unterstützt, lautet der Wert Teilweise.

Weitere Informationen zu ABAC finden Sie unter [Definieren von Berechtigungen mit ABAC-Autorisierung](#) im IAM-Benutzerhandbuch. Um ein Tutorial mit Schritten zur Einstellung von ABAC anzuzeigen, siehe [Attributbasierte Zugriffskontrolle \(ABAC\)](#) verwenden im IAM-Benutzerhandbuch.

Temporäre Anmeldeinformationen mit Amazon GameLift Streams verwenden

Unterstützt temporäre Anmeldeinformationen: Ja

Temporäre Anmeldeinformationen ermöglichen kurzfristigen Zugriff auf AWS Ressourcen und werden automatisch erstellt, wenn Sie einen Verbund verwenden oder die Rollen wechseln. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter [Temporäre Anmeldeinformationen in IAM](#) und [AWS-Services , die mit IAM funktionieren](#) im IAM-Benutzerhandbuch.

Serviceübergreifende Hauptberechtigungen für Amazon Streams GameLift

Unterstützt Forward Access Sessions (FAS): Ja

Forward Access Sessions (FAS) verwenden die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, kombiniert mit der Anfrage, Anfragen an nachgelagerte Dienste AWS-Service zu stellen. Einzelheiten zu den Richtlinien für FAS-Anforderungen finden Sie unter [Zugriffssitzungen weiterleiten](#).

Beim Erstellen neuer Anwendungsressourcen verwendet Amazon GameLift Streams die Berechtigungen des aufrufenden Prinzipals, um auf den Amazon S3 S3-Bucket zuzugreifen, der die Anwendungsdateien des Kunden enthält. Amazon GameLift Streams untersucht auch den aufrufenden Principal, um zu überprüfen, ob er für bestimmte regionsübergreifende Funktionen berechtigt ist, z. B. Streamgruppen mit mehreren Standorten.

Servicerollen für Amazon GameLift Streams

Unterstützt Servicerollen: Nein

Eine Servicerolle ist eine [IAM-Rolle](#), die ein Service annimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern

und löschen. Weitere Informationen finden Sie unter [Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service](#) im IAM-Benutzerhandbuch.

Warning

Das Ändern der Berechtigungen für eine Servicerolle kann die Funktionalität von Amazon GameLift Streams beeinträchtigen. Bearbeiten Sie Servicerollen nur, wenn Amazon GameLift Streams Sie dazu anleitet.

Servicebezogene Rollen für Amazon Streams GameLift

Unterstützt serviceverknüpfte Rollen: Ja

Eine serviceverknüpfte Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Dienstbezogene Rollen werden in Ihrem Dienst angezeigt AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

Details zum Erstellen oder Verwalten von serviceverknüpften Rollen finden Sie unter [AWS -Services, die mit IAM funktionieren](#). Suchen Sie in der Tabelle nach einem Service mit einem Yes in der Spalte Service-linked role (Serviceverknüpfte Rolle). Wählen Sie den Link Yes (Ja) aus, um die Dokumentation für die serviceverknüpfte Rolle für diesen Service anzuzeigen.

Beispiele für identitätsbasierte Richtlinien für Amazon Streams GameLift

Standardmäßig sind Benutzer und Rollen nicht berechtigt, Amazon GameLift Streams-Ressourcen zu erstellen oder zu ändern. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter [Erstellen von IAM-Richtlinien \(Konsole\)](#) im IAM-Benutzerhandbuch.

Einzelheiten zu den von Amazon GameLift Streams definierten Aktionen und Ressourcentypen, einschließlich des Formats ARNs für die einzelnen Ressourcentypen, finden Sie unter [Aktionen, Ressourcen und Bedingungsschlüssel für Amazon GameLift Streams](#) in der Service Authorization Reference.

Themen

- [Best Practices für Richtlinien](#)
- [Verwenden der Amazon GameLift Streams-Konsole](#)
- [Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer](#)

Best Practices für Richtlinien

Identitätsbasierte Richtlinien legen fest, ob jemand Amazon GameLift Streams-Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder diese löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Wenn Sie identitätsbasierte Richtlinien erstellen oder bearbeiten, befolgen Sie diese Richtlinien und Empfehlungen:

- Erste Schritte mit AWS verwalteten Richtlinien und Umstellung auf Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um damit zu beginnen, Ihren Benutzern und Workloads Berechtigungen zu gewähren. Sie sind in Ihrem verfügbar. AWS-Konto Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom AWS Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind. Weitere Informationen finden Sie unter [Von AWS verwaltete Richtlinien](#) oder [Von AWS verwaltete Richtlinien für Auftragsfunktionen](#) im IAM-Benutzerhandbuch.
- Anwendung von Berechtigungen mit den geringsten Rechten – Wenn Sie mit IAM-Richtlinien Berechtigungen festlegen, gewähren Sie nur die Berechtigungen, die für die Durchführung einer Aufgabe erforderlich sind. Sie tun dies, indem Sie die Aktionen definieren, die für bestimmte Ressourcen unter bestimmten Bedingungen durchgeführt werden können, auch bekannt als die geringsten Berechtigungen. Weitere Informationen zur Verwendung von IAM zum Anwenden von Berechtigungen finden Sie unter [Richtlinien und Berechtigungen in IAM](#) im IAM-Benutzerhandbuch.
- Verwenden von Bedingungen in IAM-Richtlinien zur weiteren Einschränkung des Zugriffs – Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen zu beschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben, um festzulegen, dass alle Anforderungen mithilfe von SSL gesendet werden müssen. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese für einen bestimmten Zweck verwendet werden AWS-Service, z. CloudFormation B. Weitere Informationen finden Sie unter [IAM-JSON-Richtlinienelemente: Bedingung](#) im IAM-Benutzerhandbuch.
- Verwenden von IAM Access Analyzer zur Validierung Ihrer IAM-Richtlinien, um sichere und funktionale Berechtigungen zu gewährleisten – IAM Access Analyzer validiert neue

und vorhandene Richtlinien, damit die Richtlinien der IAM-Richtliniensprache (JSON) und den bewährten IAM-Methoden entsprechen. IAM Access Analyzer stellt mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen zur Verfügung, damit Sie sichere und funktionale Richtlinien erstellen können. Weitere Informationen finden Sie unter [Richtlinienvalidierung mit IAM Access Analyzer](#) im IAM-Benutzerhandbuch.

- Multi-Faktor-Authentifizierung (MFA) erforderlich — Wenn Sie ein Szenario haben, das IAM-Benutzer oder einen Root-Benutzer in Ihrem System erfordert AWS-Konto, aktivieren Sie MFA für zusätzliche Sicherheit. Um MFA beim Aufrufen von API-Vorgängen anzufordern, fügen Sie Ihren Richtlinien MFA-Bedingungen hinzu. Weitere Informationen finden Sie unter [Sicherer API-Zugriff mit MFA](#) im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter [Best Practices für die Sicherheit in IAM](#) im IAM-Benutzerhandbuch.

Verwenden der Amazon GameLift Streams-Konsole

Um auf die Amazon GameLift Streams-Konsole zugreifen zu können, benötigen Sie ein Mindestmaß an Berechtigungen. Diese Berechtigungen müssen es Ihnen ermöglichen, Details zu den Amazon GameLift Streams-Ressourcen in Ihrem aufzulisten und anzuzeigen AWS-Konto. Wenn Sie eine identitätsbasierte Richtlinie erstellen, die strenger ist als die mindestens erforderlichen Berechtigungen, funktioniert die Konsole nicht wie vorgesehen für Entitäten (Benutzer oder Rollen) mit dieser Richtlinie.

Sie müssen Benutzern, die nur die API AWS CLI oder die AWS API aufrufen, keine Mindestberechtigungen für die Konsole gewähren. Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die der API-Operation entsprechen, die die Benutzer ausführen möchten.

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die IAM-Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer Benutzeridentität angefügt sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der API AWS CLI oder AWS .

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

```
{  
    "Sid": "ViewOwnUserInfo",  
    "Effect": "Allow",  
    "Action": [  
        "iam:GetUserPolicy",  
        "iam>ListGroupsForUser",  
        "iam>ListAttachedUserPolicies",  
        "iam>ListUserPolicies",  
        "iam GetUser"  
    ],  
    "Resource": ["arn:aws:iam::*:user/${aws:username}"]  
},  
{  
    "Sid": "NavigateInConsole",  
    "Effect": "Allow",  
    "Action": [  

```

Fehlerbehebung bei Identität und Zugriff auf Amazon GameLift Streams

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit Amazon GameLift Streams und IAM auftreten können.

Themen

- [Ich bin nicht berechtigt, eine Aktion in Amazon GameLift Streams durchzuführen](#)
- [Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine Amazon GameLift Streams-Ressourcen ermöglichen](#)

Ich bin nicht berechtigt, eine Aktion in Amazon GameLift Streams durchzuführen

Wenn Sie eine Fehlermeldung erhalten, dass Sie nicht zur Durchführung einer Aktion berechtigt sind, müssen Ihre Richtlinien aktualisiert werden, damit Sie die Aktion durchführen können.

Der folgende Beispielfehler tritt auf, wenn der IAM-Benutzer `mateojackson` versucht, über die Konsole Details zu einer fiktiven `my-example-widget`-Ressource anzuzeigen, jedoch nicht über `gameliftstreams:GetWidget`-Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:  
gameliftstreams:GetWidget on resource: my-example-widget
```

In diesem Fall muss die Richtlinie für den Benutzer `mateojackson` aktualisiert werden, damit er mit der `gameliftstreams:GetWidget`-Aktion auf die `my-example-widget`-Ressource zugreifen kann.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine Amazon GameLift Streams-Ressourcen ermöglichen

Dies ist mit Amazon GameLift Streams nicht möglich. Der gesamte API-Zugriff ist auf das Konto beschränkt, dem die Ressourcen gehören. Stattdessen sind Kunden, die Inhalte extern teilen möchten, dafür verantwortlich, ihr Konto zu verwenden, um neue Stream-Sitzungen im Namen anderer Benutzer, die Amazon GameLift Streams verwenden APIs, zu initiieren und die entsprechenden Verbindungsinformationen an die Webbrowser dieser externen Nutzer weiterzuleiten.

Konformitätsprüfung für Amazon GameLift Streams

Informationen darüber, ob AWS-Service ein [AWS-Services in den Geltungsbereich bestimmter Compliance-Programme fällt, finden Sie unter Umfang nach Compliance-Programm AWS-Services unter](#). Wählen Sie dort das Compliance-Programm aus, an dem Sie interessiert sind. Allgemeine Informationen finden Sie unter [AWS Compliance-Programme AWS](#).

Sie können Prüfberichte von Drittanbietern unter herunterladen AWS Artifact. Weitere Informationen finden Sie unter [Berichte herunterladen unter](#).

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services hängt von der Vertraulichkeit Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. Weitere Informationen zu Ihrer Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services finden Sie in der [AWS Sicherheitsdokumentation](#).

Resilienz in Amazon GameLift Streams

Die AWS globale Infrastruktur basiert auf Availability AWS-Regionen Zones. AWS-Regionen bieten mehrere physisch getrennte und isolierte Availability Zones, die über Netzwerke mit niedriger Latenz, hohem Durchsatz und hoher Redundanz miteinander verbunden sind. Mithilfe von Availability Zones können Sie Anwendungen und Datenbanken erstellen und ausführen, die automatisch Failover zwischen Zonen ausführen, ohne dass es zu Unterbrechungen kommt. Availability Zones sind besser verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Weitere Informationen zu Availability Zones AWS-Regionen und Availability Zones finden Sie unter [AWS Globale Infrastruktur](#).

Zusätzlich zur Datenredundanz, die durch die AWS globale Infrastruktur bereitgestellt wird, verfügt Amazon GameLift Streams über eine robuste Multi-Availability Zone-Infrastruktur. Im Falle eines Ausfalls in der Availability Zone können einzelne bestehende Sitzungen betroffen sein, aber der Service sorgt weiterhin für einen Lastenausgleich für neue Sitzungen zwischen funktionsfähigen Availability Zones.

Infrauktursicherheit in Amazon GameLift Streams

Als verwalteter Service ist Amazon GameLift Streams durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS Sicherheitsdiensten und zum AWS Schutz der Infrastruktur finden Sie unter [AWS Cloud-Sicherheit](#). Informationen zum Entwerfen Ihrer AWS Umgebung unter Verwendung der bewährten Methoden für die Infrauktursicherheit finden Sie unter [Infrastructure Protection](#) in Security Pillar AWS Well-Architected Framework.

Sie verwenden AWS veröffentlichte API-Aufrufe, um über das Netzwerk auf Amazon GameLift Streams zuzugreifen. Kunden müssen Folgendes unterstützen:

- Transport Layer Security (TLS). Wir benötigen TLS 1.2 und empfehlen TLS 1.3.

- Verschlüsselungs-Suiten mit Perfect Forward Secrecy (PFS) wie DHE (Ephemeral Diffie-Hellman) oder ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi.

Wiederverwendung und Mehrmandantenfähigkeit in Amazon Streams GameLift

Amazon GameLift Streams teilt keine Rechenressourcen zwischen Stream-Gruppen oder mit anderen AWS Kunden. Einige Amazon GameLift Streams-Stream-Gruppen verlassen sich auf die interne gemeinsame Nutzung von Ressourcen.

Wiederverwendung von Rechenressourcen

Innerhalb einer Stream-Gruppe werden Ressourcen im Laufe der Zeit wiederverwendet, um mehrere Sitzungen mit minimaler Ausfallzeit abzuwickeln. Die spezifischen Details der Wiederverwendung unterscheiden sich zwischen Windows- und Nicht-Windows-Streamgruppen.

Stream-Gruppen außerhalb von Windows mit Stream-Klassen wie gen4n_high, gen5n_ultra, gen6n_ultra oder gen6n_pro führen Ihre Anwendungen in speziellen Sitzungsspezifischen Containern aus. Jede Stream-Sitzung beginnt mit einer Kopie der Anwendungsdateien und einem leeren Benutzerprofilordner. Wenn eine Sitzung beendet wird, werden alle Änderungen am Dateisystem verworfen und alle von Ihrer Anwendung gestarteten Prozesse werden im Rahmen der Container-Bereinigung beendet.

Windows-basierte Streamgruppen mit Stream-Klassen wie gen4n_win2022, gen5n_win2022, gen6n_ultra_gen6n_win2022, oder gen6n_pro_gen6n_win2022 führen Ihre Anwendungen direkt auf dem Host-Betriebssystem aus. Jede Streamsitzung beginnt mit einer Kopie der Anwendungsdateien und einem leeren Benutzerprofilordner. Wenn eine Sitzung beendet wird, werden der Benutzerprofilordner und der Anwendungsordner vollständig zurückgesetzt. Unterprozesse, die von Ihrer Anwendung gestartet wurden, werden beendet. Wenn Ihre Anwendung Dateien außerhalb des Benutzerprofilordners und des Anwendungsordners ändert oder die Systemregistrierung ändert, können diese Änderungen über mehrere Sitzungen hinweg bestehen bleiben.

Bei jeder Stream-Gruppenkonfiguration werden die zugrunde liegenden Rechenressourcen und die Betriebssystemumgebung im Laufe der Zeit wiederverwendet, um neue Stream-Sitzungen zu starten. Im Rahmen des [Modells der gemeinsamen Verantwortung liegt](#) es in Ihrer Verantwortung, die

Sicherheit Ihrer Anwendungen zu gewährleisten und zu verhindern, dass nicht vertrauenswürdiger Code ausgeführt oder wichtige Betriebssystemdateien geändert werden.

Stream-Gruppen mit mehreren Mandanten

Streamgruppen sind entweder Single-Tenant- oder Multi-Tenant-Gruppen, je nachdem, welche Stream-Klasse Sie ausgewählt haben. Stream-Klassen mit mehreren Mandanten, z. B. gen4n_high oder gen5n_high teilen sich eine GPU für mehrere gleichzeitige Sitzungen. In diesem Zusammenhang bezieht sich Multi-Tenancy auf die Ausführung mehrerer Sitzungen gleichzeitig auf der zugrunde liegenden Hardware. Die Hardware ist immer noch für Ihre Stream-Gruppe reserviert und wird nicht von mehreren Stream-Gruppen oder mit anderen AWS Kunden gemeinsam genutzt.

Dieses Mehrmandanten-Stream-Gruppenmodell ist einzigartig für Amazon GameLift Streams und hat wichtige Auswirkungen auf Sicherheit und Leistung. Die Sicherheitslage einer Multi-Tenant-Stream-Gruppe entspricht dem Hosten mehrerer Anwendungscontainer auf einem einzigen physischen Server. Diese Sicherheitslage ist nicht von Natur aus unsicher, kann aber die Auswirkungen vorhandener Sicherheitslücken in Ihren Anwendungen verstärken. Im Rahmen des [Modells der gemeinsamen Verantwortung liegt](#) es in Ihrer Verantwortung, die Sicherheit Ihrer Anwendungen aufrechtzuerhalten.

Amazon GameLift Streams bemüht sich sicherzustellen, dass sich Sitzungen mit mehreren Mandanten nicht gegenseitig stören. Wenn eine Anwendung jedoch CPU- oder GPU-Ressourcen verbraucht, ohne die definierten Grenzwerte der Stream-Klasse zu berücksichtigen, kann dies Auswirkungen auf andere Streams haben, die versuchen, dieselben gemeinsam genutzten Ressourcen zu verwenden. In einer Gruppe mit „hohem“ Datenstrom und zwei Mandanten pro GPU kann sich eine gierige Anwendung beispielsweise negativ auf einen anderen Stream auswirken. Ihre Anwendung sollte ihren eigenen Ressourcenverbrauch regulieren. Wenn sich Ihre Anwendung nicht selbst regulieren kann und Ihr Anwendungsfall keine Toleranz für mögliche Leistungsschwankungen beim „Noisy Neighbor“ bietet, wird eine Stream-Klasse mit einem Mandanten wie gen5n_win2022, gen6n_pro_win2022 gen5n_ultragen6n_ultra, oder empfohlen.

VPC-Endpunkte in Amazon Streams verbinden GameLift

Sie können die Sicherheitslage Ihrer VPC verbessern, indem Sie Amazon GameLift Streams so konfigurieren, dass es einen VPC-Endpunkt mit Schnittstelle verwendet. Schnittstellenendpunkte werden von einer Technologie unterstützt AWS PrivateLink, mit der Sie privat auf Amazon GameLift Streams zugreifen können, APIs indem Sie private IP-Adressen verwenden. AWS PrivateLink schränkt den gesamten Netzwerkverkehr zwischen Ihrer VPC und Amazon GameLift Streams auf das

Amazon-Netzwerk ein. Sie benötigen kein Internet-Gateway, kein NAT-Gerät und kein Virtual Private Gateway.

Weitere Informationen zu AWS PrivateLink VPC-Endpunkten finden Sie unter [VPC-Endpunkte](#) im Amazon VPC-Benutzerhandbuch.

 Note

AWS PrivateLink gilt nur für API-Endpunkte. Von Amazon GameLift Streams verwaltete Stream-Sitzungen verwenden immer öffentliche Netzwerkadressen.

VPC-Endpunkte für Amazon Streams erstellen GameLift

Um den VPC-Endpunkt für den Amazon GameLift Streams-Service zu erstellen, verwenden Sie das Verfahren [Access an AWS service using an interface VPC-Endpoint](#) im Amazon VPC-Benutzerhandbuch, um den folgenden Endpunkt zu erstellen:

- com.amazonaws.*region*.gameliftstreams

 Note

region steht für die Regionskennung für eine von Amazon GameLift Streams AWS-Region unterstützte Region, z. B. us-east-2 für die Region USA Ost (Ohio).

Erstellen einer VPC-Endpunktrichtlinie für Amazon Streams GameLift

Sie können Ihrem VPC-Endpunkt eine Endpunktrichtlinie hinzufügen, die den Zugriff auf Amazon GameLift Streams steuert. Die Richtlinie gibt die folgenden Informationen an:

- Prinzipal, der die Aktionen ausführen kann.
- Aktionen, die ausgeführt werden können
- Die Ressourcen, für die Aktionen ausgeführt werden können.

Weitere Informationen finden Sie unter [Steuern des Zugriffs auf VPC-Endpunkte mithilfe von Endpunktrichtlinien](#) im Amazon-VPC-Benutzerhandbuch.

Example Beispiel: VPC-Endpunktrichtlinie für Amazon Streams GameLift

Das Folgende ist ein Beispiel für eine Endpunktrichtlinie für Amazon GameLift Streams. Wenn diese Richtlinie an einen Endpunkt angehängt ist, gewährt sie die Erlaubnis, Stream-Gruppen zu erstellen und aufzulisten.

```
{  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Principal": "*",  
            "Action": [  
                "gameliftstreams:CreateStreamGroup",  
                "gameliftstreams>ListStreamGroups"  
            ],  
            "Resource": [  
                "*"  
            ]  
        }  
    ]  
}
```

Konfiguration und Schwachstellenanalyse in Amazon GameLift Streams

Konfiguration und IT-Steuerung liegen in der gemeinsamen Verantwortung AWS von Ihnen, unserem Kunden. Weitere Informationen finden Sie im [Modell der AWS gemeinsamen Verantwortung](#). AWS kümmert sich um grundlegende Sicherheitsaufgaben wie das Patchen von Gastbetriebssystemen (OS) und Datenbanken, die Firewallkonfiguration und die Notfallwiederherstellung. Diese Verfahren wurden von qualifizierten Dritten überprüft und zertifiziert. Weitere Informationen finden Sie in der folgenden Ressource: [Amazon Web Services: Überblick über Sicherheitsprozesse](#) (Whitepaper).

Die folgenden bewährten Sicherheitsmethoden beziehen sich auch auf die Konfiguration und Schwachstellenanalyse in Amazon GameLift Streams:

- Kunden sind für die Verwaltung der Software verantwortlich, die in Amazon GameLift Streams-Stream-Gruppen für das Stream-Hosting bereitgestellt wird. Das heißt:

- Vom Kunden bereitgestellte Anwendungsinhalte und Software, einschließlich Updates und Sicherheitspatches, sollten beibehalten werden. Um zu aktualisieren, erstellen Sie eine neue Amazon GameLift Streams-Anwendung und stellen Sie sie für neue Stream-Gruppen bereit.
- Derzeit werden das Betriebssystem und die Laufzeitumgebung für eine Stream-Gruppe nur aktualisiert, wenn Sie eine neue Stream-Gruppe erstellen. Um das Betriebssystem und andere Anwendungen, die Teil der Laufzeitumgebung sind, zu patchen, zu aktualisieren und zu sichern, empfehlen wir, die Stream-Gruppen unabhängig von Anwendungsupdates alle zwei bis vier Wochen zu recyceln.
- Kunden sollten erwägen, ihre Spiele regelmäßig mit den neuesten SDK-Versionen zu aktualisieren, einschließlich des AWS SDK und des Amazon GameLift Streams Web Client SDK.

Bewährte Sicherheitsmethoden für Amazon GameLift Streams

Amazon GameLift Streams bietet eine Reihe von Sicherheitsfunktionen, die Sie bei der Entwicklung und Implementierung Ihrer eigenen Sicherheitsrichtlinien berücksichtigen sollten. Die folgenden bewährten Methoden stellen allgemeine Richtlinien und keine vollständige Sicherheitslösung dar. Da diese bewährten Methoden für Ihre Umgebung möglicherweise nicht angemessen oder ausreichend sind, sollten Sie sie als hilfreiche Überlegungen und nicht als bindend ansehen.

- Derzeit werden das Betriebssystem und die Laufzeitumgebung für eine Stream-Gruppe nur aktualisiert, wenn Sie eine neue Stream-Gruppe erstellen. Um das Betriebssystem und andere Anwendungen, die Teil der Laufzeitumgebung sind, zu patchen, zu aktualisieren und zu sichern, empfehlen wir, die Stream-Gruppen unabhängig von Anwendungsupdates alle zwei bis vier Wochen zu recyceln.
- [Bewährte Methoden für Sicherheit, Identität und Compliance](#)

Überwachung von Amazon GameLift Streams

Die Überwachung ist ein wichtiger Bestandteil der Aufrechterhaltung der Zuverlässigkeit, Verfügbarkeit und Leistung von Amazon GameLift Streams und Ihren anderen AWS Lösungen. AWS bietet die folgenden Überwachungstools, um Amazon GameLift Streams anzusehen, zu melden, wenn etwas nicht stimmt, und gegebenenfalls automatische Maßnahmen zu ergreifen:

- Amazon CloudWatch überwacht Ihre AWS Ressourcen und die Anwendungen, auf denen Sie laufen, AWS in Echtzeit. Sie können Kennzahlen erfassen und verfolgen, benutzerdefinierte Dashboards erstellen und Alarne festlegen, die Sie benachrichtigen oder Maßnahmen ergreifen, wenn eine bestimmte Metrik einen von Ihnen festgelegten Schwellenwert erreicht. Weitere Informationen finden Sie im [CloudWatch Amazon-Benutzerhandbuch](#).
- Mit Amazon CloudWatch Logs können Sie Ihre Protokolldateien von Diensten wie Amazon Elastic Compute Cloud und anderen Quellen überwachen AWS CloudTrail, speichern und darauf zugreifen. CloudWatch Logs können Informationen in den Protokolldateien überwachen und Sie benachrichtigen, wenn Ihre Services bestimmte Schwellenwerte erreichen. Sie können Ihre Protokolldaten auch in einem sehr robusten Speicher archivieren. Weitere Informationen finden Sie im [Amazon CloudWatch Logs-Benutzerhandbuch](#).
- AWS CloudTrail erfasst API-Aufrufe und zugehörige Ereignisse, die von oder im Namen Ihres AWS Kontos getätigt wurden, und übermittelt die Protokolldateien an einen von Ihnen angegebenen Amazon Simple Storage Service-Bucket. Sie können feststellen, welche Benutzer und Konten angerufen wurden AWS, von welcher Quell-IP-Adresse aus die Aufrufe getätigt wurden und wann die Aufrufe erfolgten. Weitere Informationen finden Sie im [AWS CloudTrail -Benutzerhandbuch](#).
- Leistungsstatistiken in Echtzeit erfassen während Stream-Sitzungen Leistungsstatistiken auf Anwendungsebene und auf gemeinsam genutzter Systemebene. Sie können diese Statistiken in Echtzeit auf dem Client oder nach der Sitzung als CSV-Datei in exportierten Sitzungsdateien abrufen. Mit dieser Funktion können Sie die CPU-, Arbeitsspeicher-, GPU- und VRAM-Auslastung Ihres Streams überwachen. Weitere Informationen finden Sie unter [the section called "Leistungsstatistiken in Echtzeit"](#).

Überwachen Sie Amazon GameLift Streams mit Amazon CloudWatch

Sie können Amazon GameLift Streams mithilfe von Amazon Streams überwachen CloudWatch, das Rohdaten sammelt und zu lesbaren Kennzahlen verarbeitet, die nahezu in Echtzeit verfügbar

sind. Diese Statistiken werden 15 Monate gespeichert, damit Sie auf Verlaufsinformationen zugreifen können und einen besseren Überblick darüber erhalten, wie Ihre Webanwendung oder der Service ausgeführt werden. Sie können auch Alarme einrichten, die auf bestimmte Grenzwerte achten und Benachrichtigungen senden oder Aktivitäten auslösen, wenn diese Grenzwerte erreicht werden.

Weitere Informationen finden Sie im [CloudWatch Amazon-Benutzerhandbuch](#).

Amazon GameLift Streams bietet Kennzahlen, mit denen Kunden Folgendes überwachen können:

- Kapazität und Nutzung der Stream-Gruppe.
- Stream-Leistung und Ressourcennutzung.
- Streamen Sie den Status, um Probleme zu lösen und Benutzer zu unterstützen.
- Kundenbindung bei allen Inhaltsangeboten.
- Nutzung von Datenkanälen.

In den folgenden Tabellen sind die Dimensionen und Metriken für Amazon GameLift Streams aufgeführt.

Kapazität und Nutzung der Stream-Gruppe

Verwenden Sie diese Kennzahlen, um Ressourcen bedarfsgerecht zu skalieren. Diese Kennzahlen werden jede Minute veröffentlicht.

Important

Für Stream-Gruppen, die vor dem 5. September 2025 erstellt wurden

Aufgrund eines Problems mit CloudWatch der Datenaufbewahrungsrichtlinie sind genaue Kapazitätskennzahlen nur für die letzten 15 Tage verfügbar. Bei Kapazitätskennzahlen, die älter als 15 Tage sind, sind keine Daten sichtbar, wenn der Zeitraum 1 Minute beträgt, und die angezeigten Daten sind ungenau, wenn der Zeitraum 5 Minuten oder mehr beträgt.

Um dieses Problem zu umgehen, können Sie einer Summenstatistik in Ihrem CloudWatch Diagramm $SUM(METRICS())/5$ mathematische Berechnungen hinzufügen (z. B. wenn Sie einen Zeitraum von 5 Minuten verwenden), um das Problem zu umgehen, um genaue Kapazitätszahlen zu ermitteln, die über die Aufbewahrungsfrist von 15 Tagen und 1 Minute hinausgehen.

Um dieses Problem zu beheben, erstellen Sie Ihre Stream-Gruppen neu.

Metrik	Description	Dimension	Einheit
ActiveCapacity	Die Anzahl der Rechenressourcen, die bereitgestellt wurden und bereit sind, gestreamt zu werden. Dazu gehören Ressourcen, die gerade gestreamt werden, und Ressourcen, die inaktiv sind und bereit sind, auf neue Stream-Anfragen zu antworten.	(StreamGr oupld, Standort)	Anzahl
IdleCapacity	Der numerische Teil der aktiven Kapazität, der derzeit nicht gestreamt wird. Er stellt die Verfügbarkeit von Rechenressourcen dar, um auf neue Stream-Anfragen zu antworten.	(StreamGr oupld, Standort)	Anzahl

Leistung und Ressourcennutzung der Stream-Gruppe

Diese Kennzahlen werden jede Minute veröffentlicht.

Metrik	Description	Dimension	Einheit
MemoryUtilization	% des verfügbaren Speichers, der vom Stream verwendet wird.	(StreamGr oupld, Standort), (ApplicationId, StreamClass)	Prozentsatz
CPUUtilization	% der verfügbaren CPU, die vom Stream verwendet wird.	(StreamGr oupld,	Prozentsatz

Metrik	Description	Dimension	Einheit
		Standort), (ApplicationId, StreamClass)	
FrameCaptureRate	Geschwindigkeit, mit der Frames von der Anwendung erfasst werden.	(StreamGroupId, Ort), (ApplicationId, StreamClass)	Keine
AudioCaptureRate	Geschwindigkeit, mit der Hörproben aus der Anwendung aufgenommen werden.	(StreamGroupId, Ort), (ApplicationId, StreamClass)	Keine
RoundTripTime	Hin- und Rückflugzeit zwischen Client und Server.	(StreamGroupId, Standort), (ApplicationId, StreamClass)	ms

Status des Streams

Diese Metriken werden am Ende einer Stream-Sitzung veröffentlicht.

Metrik	Description	Dimension	Einheit
TerminatedStreamSessions	Anzahl der beendeten Sitzungen im Bundesstaat TERMINATED	(StreamGroupId, Ort), (ApplicationId, StreamClass)	Anzahl

Metrik	Description	Dimension	Einheit
ErroredStreamSessions	Anzahl der im Bundesstaat beendeten Sitzungen ERROR	(StreamGr oupld, Ort), (Applicat ionId, StreamClass)	Anzahl

Kundenbindung

Diese Metriken werden am Ende einer Stream-Sitzung veröffentlicht.

Metrik	Description	Dimension	Einheit
Länge der Sitzung	Dauer der Stream-Sitzung	(StreamGr oupld, Ort), (Applicat ionId, StreamClass)	Sekunden

Datenkanäle

Diese Metriken werden am Ende einer Stream-Sitzung veröffentlicht.

Metrik	Description	Dimension	Einheit
DataChannel-ApplicationConnected	Häufigkeit, mit der Ihre Anwendung eine Verbindung zum Datenkanalport herstellt. Diese Anzahl beträgt höchstens 1 pro Stream-Sitzung.	(StreamGr oupld, Ort), (Applicat ionId, StreamClass)	Anzahl
DataChannel-ApplicationMessage	Anzahl der Nachrichten, die Ihre Anwendung an Ihren Kunden gesendet hat.	(StreamGr oupld, Standort)	Anzahl

Metrik	Description	Dimension	Einheit
		, (ApplicationId, StreamClass)	
DataChannel-ApplicationMessageBytes	Gesamtzahl der Nachrichten, die Ihre Anwendung an Ihren Client gesendet hat.	(StreamGroupID, Standort), (ApplicationId, StreamClass)	Bytes
DataChannel-ClientMessage	Anzahl der Nachrichten, die Ihr Kunde an Ihre Anwendung gesendet hat.	(StreamGroupID, Standort), (ApplicationId, StreamClass)	Anzahl
DataChannel-ClientMessageBytes	Gesamtzahl der Byte der Nachrichten, die Ihr Client an Ihre Anwendung gesendet hat.	(StreamGroupID, Standort), (ApplicationId, StreamClass)	Bytes

Protokollieren von Amazon GameLift Streams-API-Aufrufen mit AWS CloudTrail

Amazon GameLift Streams ist in einen Service integriert [AWS CloudTrail](#), der eine Aufzeichnung der von einem Benutzer, einer Rolle oder einem ausgeführten Aktionen bereitstellt AWS-Service. CloudTrail erfasst alle API-Aufrufe für Amazon GameLift Streams als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der Amazon GameLift Streams-Konsole und Code-Aufrufe der Amazon GameLift Streams-API-Operationen. Anhand der von gesammelten Informationen können Sie die

Anfrage CloudTrail, die an Amazon GameLift Streams gestellt wurde, die IP-Adresse, von der aus die Anfrage gestellt wurde, den Zeitpunkt der Anfrage und weitere Details ermitteln.

Jeder Ereignis- oder Protokolleintrag enthält Informationen zu dem Benutzer, der die Anforderung generiert hat. Die Identitätsinformationen unterstützen Sie bei der Ermittlung der folgenden Punkte:

- Ob die Anfrage mit Anmeldeinformationen des Root-Benutzers oder des Benutzers gestellt wurde.
- Die Anforderung wurde im Namen eines IAM-Identity-Center-Benutzers erstellt.
- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anforderung aus einem anderen AWS-Service gesendet wurde.

CloudTrail ist in Ihrem aktiv AWS-Konto , wenn Sie das Konto erstellen, und Sie haben automatisch Zugriff auf den CloudTrail Event-Verlauf. Der CloudTrail Ereignisverlauf bietet eine einsehbare, durchsuchbare, herunterladbare und unveränderliche Aufzeichnung der aufgezeichneten Verwaltungsereignisse der letzten 90 Tage in einem AWS-Region. Weitere Informationen finden Sie im AWS CloudTrail Benutzerhandbuch unter [Arbeiten mit dem CloudTrail Ereignisverlauf](#). Für die Anzeige des Eventverlaufs CloudTrail fallen keine Gebühren an.

Für eine fortlaufende Aufzeichnung der Ereignisse in AWS-Konto den letzten 90 Tagen erstellen Sie einen Trail- oder [CloudTrail Lake-Event-Datenspeicher](#).

CloudTrail Pfade

Ein Trail ermöglicht CloudTrail die Übermittlung von Protokolldateien an einen Amazon S3 S3-Bucket. Alle mit dem erstellten Pfade AWS-Managementkonsole sind regionsübergreifend. Sie können mithilfe von AWS CLI einen Einzel-Region- oder einen Multi-Region-Trail erstellen. Es wird empfohlen, einen Trail mit mehreren Regionen zu erstellen, da Sie alle Aktivitäten AWS-Regionen in Ihrem Konto erfassen. Wenn Sie einen Einzel-Region-Trail erstellen, können Sie nur die Ereignisse anzeigen, die im AWS-Region des Trails protokolliert wurden. Weitere Informationen zu Trails finden Sie unter [Erstellen eines Trails für Ihr AWS-Konto](#) und [Erstellen eines Trails für eine Organisation](#) im AWS CloudTrail -Benutzerhandbuch.

Sie können eine Kopie Ihrer laufenden Verwaltungsereignisse kostenlos an Ihren Amazon S3 S3-Bucket senden, CloudTrail indem Sie einen Trail erstellen. Es fallen jedoch Amazon S3 S3-Speichergebühren an. Weitere Informationen zur CloudTrail Preisgestaltung finden Sie unter [AWS CloudTrail Preise](#). Informationen zu Amazon-S3-Preisen finden Sie unter [Amazon S3 – Preise](#).

CloudTrail Datenspeicher für Ereignisse in Lake

CloudTrail Mit Lake können Sie SQL-basierte Abfragen für Ihre Ereignisse ausführen. CloudTrail [Lake konvertiert bestehende Ereignisse im zeilenbasierten JSON-Format in das Apache ORC-Format](#). ORC ist ein spaltenförmiges Speicherformat, das für den schnellen Abruf von Daten optimiert ist. Die Ereignisse werden in Ereignisdatenspeichern zusammengefasst, bei denen es sich um unveränderliche Sammlungen von Ereignissen handelt, die auf Kriterien basieren, die Sie mit Hilfe von [erweiterten Ereignisselektoren](#) auswählen. Die Selektoren, die Sie auf einen Ereignisdatenspeicher anwenden, steuern, welche Ereignisse bestehen bleiben und für Sie zur Abfrage verfügbar sind. Weitere Informationen zu CloudTrail Lake finden Sie unter [Arbeiten mit AWS CloudTrail Lake](#) im AWS CloudTrail Benutzerhandbuch.

CloudTrail Für das Speichern und Abfragen von Ereignisdaten in Lake fallen Kosten an. Beim Erstellen eines Ereignisdatenspeichers wählen Sie die [Preisoption](#) aus, die für den Ereignisdatenspeicher genutzt werden soll. Die Preisoption bestimmt die Kosten für die Erfassung und Speicherung von Ereignissen sowie die standardmäßige und maximale Aufbewahrungszeit für den Ereignisdatenspeicher. Weitere Informationen zur CloudTrail Preisgestaltung finden Sie unter [AWS CloudTrail Preise](#).

Amazon GameLift Streams-Datenereignisse in CloudTrail

[Datenereignisse](#) liefern Informationen über die Ressourcenoperationen, die auf oder in einer Ressource ausgeführt werden (z. B. das Starten einer Stream-Sitzung in einer Stream-Gruppe). Sie werden auch als Vorgänge auf Datenebene bezeichnet. Datenereignisse sind oft Aktivitäten mit hohem Volume. Protokolliert standardmäßig CloudTrail keine Datenereignisse. Der CloudTrail Ereignisverlauf zeichnet keine Datenereignisse auf.

Für Datenereignisse werden zusätzliche Gebühren fällig. Weitere Informationen zur CloudTrail Preisgestaltung finden Sie unter [AWS CloudTrail Preisgestaltung](#).

Sie können Datenereignisse für die Amazon GameLift Streams-Ressourcentypen mithilfe der CloudTrail Konsole oder CloudTrail API-Operationen protokollieren. AWS CLI Weitere Informationen zum Protokollieren von Datenereignissen finden Sie unter [Protokollieren von Datenereignissen mit dem AWS-Managementkonsole](#) und [Protokollieren von Datenereignissen mit dem AWS Command Line Interface](#) im AWS CloudTrail -Benutzerhandbuch.

In der folgenden Tabelle sind die Amazon GameLift Streams-Ressourcentypen aufgeführt, für die Sie Datenereignisse protokollieren können. In der Spalte Ressourcentyp (Konsole) wird der Wert angezeigt, den Sie in der Liste Ressourcentyp auf der CloudTrail Konsole auswählen können. In der

Wertspalte `resources.type` wird der `resources.type` Wert angezeigt, den Sie bei der Konfiguration erweiterter Event-Selektoren mithilfe von oder angeben würden. AWS CLI CloudTrail APIs In der CloudTrail Spalte APIs Protokollierte Daten werden die API-Aufrufe angezeigt, die CloudTrail für den Ressourcentyp protokolliert wurden.

Ressourcentyp (Konsole)	<code>resources.type</code> -Wert	Daten APIs wurden protokolliert CloudTrail
GameLift Streams-Anwendung	<code>AWS::GameLiftStreams::Application</code>	<ul style="list-style-type: none"> • StartStreamSession
GameLift Stream-Gruppe streamt	<code>AWS::GameLiftStreams::StreamGroup</code>	<ul style="list-style-type: none"> • CreateStreamSessionConnection • ExportStreamSessionFiles • GetStreamSession • ListStreamSessions • ListStreamSessionsByAccount • StartStreamSession • TerminateStreamSession

Sie können erweiterte Event-Selektoren so konfigurieren, dass sie nach den Feldern `eventName`, `readOnly` und `resources.ARN` filtern, sodass nur die Ereignisse protokolliert werden, die für Sie wichtig sind. Weitere Informationen zu diesen Kontingenzen finden Sie unter [AdvancedFieldSelector](#) in der AWS CloudTrail -API-Referenz.

Amazon GameLift Streams-Verwaltungsereignisse in CloudTrail

[Verwaltungsereignisse](#) bieten Informationen über Verwaltungsvorgänge, die mit Ressourcen in Ihrem System ausgeführt werden AWS-Konto. Sie werden auch als Vorgänge auf Steuerebene bezeichnet. CloudTrail Protokolliert standardmäßig Verwaltungsereignisse.

Amazon GameLift Streams protokolliert die folgenden Vorgänge auf der Amazon GameLift Streams-Kontrollebene CloudTrail als Verwaltungsereignisse.

- [AddStreamGroupLocations](#)

- [AssociateApplications](#)
- [CreateApplication](#)
- [CreateStreamGroup](#)
- [DeleteApplication](#)
- [DeleteStreamGroup](#)
- [DisassociateApplications](#)
- [GetApplication](#)
- [GetStreamGroup](#)
- [ListApplications](#)
- [ListStreamGroups](#)
- [ListTagsForResource](#)
- [RemoveStreamGroupLocations](#)
- [TagResource](#)
- [UntagResource](#)
- [UpdateApplication](#)
- [UpdateStreamGroup](#)

Beispiele für Amazon GameLift Streams-Ereignisse

Ein Ereignis stellt eine einzelne Anfrage aus einer beliebigen Quelle dar und enthält Informationen über den angeforderten API-Vorgang, Datum und Uhrzeit des Vorgangs, Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordneter Stack-Trace der öffentlichen API-Aufrufe, sodass Ereignisse nicht in einer bestimmten Reihenfolge angezeigt werden.

Das folgende Beispiel zeigt ein CloudTrail Verwaltungsereignis, das den [CreateApplication](#)-Vorgang demonstriert.

```
{  
    "eventVersion": "1.09",  
    "userIdentity": {  
        "type": "AssumedRole",  
        "principalId": "AROA123456789EXAMPLE:assume-temporary-gameliftstreams-access-role",  
        "arn": "arn:aws:sts::111122223333:assumed-role/GameLiftStreamsTestRole/assume-temporary-gameliftstreams-access-role",  
        "accountId": "111122223333",  
        "accessKeyId": "ASIAQ1W5DZP5TJGKXH7",  
        "sessionContext": {  
            "attributes": {},  
            "sessionIssuer": {  
                "type": "AWS",  
                "principalId": "AROA123456789EXAMPLE",  
                "arn": "arn:aws:iam::111122223333:root",  
                "accountId": "111122223333",  
                "username": "root",  
                "accessKeyId": "AKIAQ1W5DZP5TJGKXH7",  
                "secretAccessKey": "XXXXXXXXXXXXXX",  
                "sessionToken": "XXXXXXXXXXXXXX",  
                "duration": 3600  
            },  
            "contextParameters": {}  
        }  
    },  
    "version": "2012-10-17",  
    "region": "us-east-1",  
    "sourceService": "GameLift",  
    "sourceRegion": "us-east-1",  
    "awsRegion": "us-east-1",  
    "eventName": "CreateApplication",  
    "userId": "AROA123456789EXAMPLE",  
    "eventSource": "aws.gamelift",  
    "eventTime": "2018-05-15T14:45:10.000Z",  
    "awsRequestID": "12345678901234567890123456789012",  
    "eventID": "12345678901234567890123456789012",  
    "eventType": "AwsEvent",  
    "recipientAccountId": "111122223333",  
    "resources": [{}],  
    "awsResponse": {}  
}
```

```
"accountId": "111122223333",
"accessKeyId": "ASIAIOSFODNN7EXAMPLE",
"sessionContext": {
    "sessionIssuer": {
        "type": "Role",
        "principalId": "AROA123456789EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/GameLiftStreamsTestRole",
        "accountId": "111122223333",
        "userName": "GameLiftStreamsTestRole"
    },
    "webIdFederationData": {},
    "attributes": {
        "creationDate": "2025-07-23T21:18:19Z",
        "mfaAuthenticated": "false"
    }
},
"eventTime": "2025-07-23T21:58:54Z",
"eventSource": "gameliftstreams.amazonaws.com",
"eventName": "CreateApplication",
"awsRegion": "us-west-2",
"sourceIPAddress": "203.0.113.0",
"userAgent": "aws-sdk-javascript/2.0.0 Linux/4.14.291-218.527.amzn2.x86_64
OpenJDK_64-Bit_Server_VM/11.0.17+9-LTS Java/11.0.17 vendor/Amazon.com_Inc. exec-env/
AWS_ECS_FARGATE io/sync http/Apache cfg/retry-mode/legacy",
"requestParameters": {
    "ApplicationSourceUri": "s3://amzn-s3-demo-bucket/MyGame",
    "Description": "MyGame canary - Proton 8",
    "RuntimeEnvironment": {
        "Type": "PROTON",
        "Version": "20230704"
    },
    "ClientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "ExecutablePath": "MyGame100.exe"
},
"responseElements": {
    "Status": "INITIALIZED",
    "ApplicationSourceUri": "s3://amzn-s3-demo-bucket/MyGame",
    "Description": "MyGame canary - Proton 8",
    "RuntimeEnvironment": {
        "Type": "PROTON",
        "Version": "20230704"
    },
    "LastUpdatedAt": 1753307934.293,
```

```
        "CreatedAt": 1753307934.293,
        "Id": "a-9ZY8X7Wv6",
        "Arn": "arn:aws:gameliftstreams:us-west-2:111122223333:application/
a-9ZY8X7Wv6",
        "ExecutablePath": "MyGame100.exe"
    },
    "requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbbb",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

Das folgende Beispiel zeigt ein CloudTrail Datenereignis aus einem Traillog, das den [StartStreamSession](#)Vorgang demonstriert.

```
{
    "Records": [
        {
            "eventVersion": "1.09",
            "userIdentity": {
                "type": "AssumedRole",
                "principalId": "AROA123456789EXAMPLE:assume-temporary-gameliftstreams-
access-role",
                "arn": "arn:aws:sts::111122223333:assumed-role/GameLiftStreamsTestRole/
assume-temporary-gameliftstreams-access-role",
                "accountId": "111122223333",
                "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
                "sessionContext": {
                    "sessionIssuer": {
                        "type": "Role",
                        "principalId": "AROA123456789EXAMPLE",
                        "arn": "arn:aws:iam::111122223333:role/
GameLiftStreamsTestRole",
                        "accountId": "111122223333",
                        "userName": "GameLiftStreamsTestRole"
                    },
                    "attributes": {
                        "creationDate": "2025-07-23T21:18:19Z",
                        "mfaAuthenticated": "false"
                    }
                }
            }
        }
    ]
}
```

```
        },
        "eventTime": "2025-07-23T23:43:46Z",
        "eventSource": "gameliftstreams.amazonaws.com",
        "eventName": "StartStreamSession",
        "awsRegion": "us-east-2",
        "sourceIPAddress": "203.0.113.0",
        "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36",
        "requestParameters": {
            "Identifier": "sg-1AB2C3De4",
            "Description": "StreamGroup sg-1AB2C3De4 Application a-9ZY8X7Wv6 Console stream",
            "AdditionalLaunchArgs": [],
            "UserId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Locations": [
                "us-east-2"
            ],
            "SignalRequest": "***",
            "Protocol": "WebRTC",
            "ApplicationIdentifier": "a-9ZY8X7Wv6",
            "ClientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "ConnectionTimeoutSeconds": 100,
            "AdditionalEnvironmentVariables": {}
        },
        "responseElements": {
            "Status": "ACTIVATING",
            "ApplicationArn": "arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6",
            "Description": "StreamGroup sg-1AB2C3De4 Application a-9ZY8X7Wv6 Console stream",
            "LastUpdatedAt": 1.753314225925E9,
            "CreatedAt": 1.753314225925E9,
            "AdditionalEnvironmentVariables": {},
            "ConnectionTimeoutSeconds": 100,
            "AdditionalLaunchArgs": [],
            "StreamGroupId": "sg-1AB2C3De4",
            "UserId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "SessionLengthSeconds": 43200,
            "SignalRequest": "***",
            "Arn": "arn:aws:gameliftstreams:us-west-2:111122223333:streamsession/sg-1AB2C3De4/ABC123def4567",
            "Protocol": "WebRTC",
        }
    }
}
```

```
        "WebSdkProtocolUrl": "https://123456789012.cloudfront.net/e3b0c44298fc1c149afb4c8996fb92427ae41e4649b934ca495991b7852b855.js",
    },
    "requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaaa",
    "eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbbb",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::GameLiftStreams::StreamGroup",
            "ARN": "arn:aws:gameliftstreams:us-west-2:111122223333:streamgroup/sg-1AB2C3De4"
        },
        {
            "accountId": "111122223333",
            "type": "AWS::GameLiftStreams::Application",
            "ARN": "arn:aws:gameliftstreams:us-west-2:111122223333:application/a-9ZY8X7Wv6"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "111122223333",
    "eventCategory": "Data"
}
]
```

Informationen zu CloudTrail Datensatzinhalten finden Sie im AWS CloudTrail Benutzerhandbuch unter [CloudTrail Datensatzinhalte](#).

Leistungsstatistiken in Echtzeit

Amazon GameLift Streams sammelt Leistungsstatistiken während aktiver Stream-Sitzungen und misst jede Sekunde die Ressourcennutzung. Verwenden Sie diese Statistiken, um die Leistung Ihrer Anwendung zu überwachen, Ressourcenengpässe zu identifizieren und Ihr Streaming-Erlebnis zu optimieren.

Zu den Leistungsstatistiken gehören sowohl Statistiken auf Anwendungsebene (CPU- und Speicherauslastung für Ihre spezifische Anwendung) als auch Statistiken auf Systemebene (CPU-, Arbeitsspeicher-, GPU- und VRAM-Auslastung für die gemeinsam genutzte Recheninfrastruktur).

Sie können Leistungsstatistiken auf zwei Arten abrufen:

- Während der Sitzung in Echtzeit: Verwenden Sie das Amazon GameLift Streams Web SDK, um Statistiken zu erhalten, sobald sie erfasst werden. Auf diese Weise können Sie Performance-Overlays erstellen und die Ressourcennutzung überwachen, während Sie mit der Anwendung interagieren.
- Nach der Sitzung als CSV-Datei: Wenn Sie Sitzungsdateien exportieren, sind die Statistiken als `stats/perf_stats_v1.csv` enthalten. Dies bietet einen vollständigen Datensatz für die Analyse und das Debugging nach der Sitzung.

Erhalten Sie Leistungsstatistiken

Empfangen Sie Statistiken in Echtzeit

Um während einer aktiven Sitzung Leistungsstatistiken in Ihrer Client-Anwendung zu erhalten, setzen Sie den `SharedWithClient` Parameter `true` beim Aufrufen der `StartStreamSession` API auf. Das Amazon GameLift Streams Web SDK bietet einen `performanceStats` Callback, der ausgelöst wird, wenn neue Statistiken aus der Streaming-Sitzung eintreffen.

 **Warning**

Nicht `SharedWithClient` für Produktionssitzungen mit Endbenutzern aktivieren. Aktivieren Sie es nur, wenn der Client vertrauenswürdig ist, z. B. für internes Debuggen und Testen.

Stellen `clientConnection.performanceStats` Sie bei der Initialisierung des Amazon GameLift Streams Web SDK eine Callback-Funktion ein, die Leistungsstatistiken empfängt.

```
const gls = new gameliftstreams.GameLiftStreams({
    videoElement: document.getElementById('streamVideoElement'),
    audioElement: document.getElementById('streamAudioElement'),
    inputConfiguration: {
        ...
    },
    clientConnection: {
        ...
        performanceStats: (perfStats) => {
            // Your callback logic here
            console.log('CPU: ' + perfStats.application.cpuNormalized);
            console.log('Memory: ' + perfStats.application.memoryMB + ' MB');
            console.log('GPU: ' + perfStats.system.gpuPercent + '%');
        },
    }
});
```

Der Callback empfängt ein `PerformanceStats` Objekt, das sowohl Statistiken auf Anwendungs- als auch auf Systemebene enthält. Einzelheiten zur Schnittstellenstruktur finden Sie in der Amazon GameLift Streams Web SDK-Dokumentation auf der [Produktseite Getting Started](#).

Die Amazon GameLift Streams-Konsole enthält auch ein integriertes Performance-Overlay, wenn Sie die Teststream-Funktion verwenden, sodass Sie Statistiken in Echtzeit überwachen können, ohne dass Implementierungsaufwand erforderlich ist.

Sie können Leistungsstatistiken mit WebRTC-Statistiken kombinieren, die von den `getAudioRTCStats()` Funktionen `getVideoRTCStats()` und im Amazon GameLift Streams Web SDK bereitgestellt werden. Diese Kombination bietet ein vollständiges Bild der Streaming-Leistung, einschließlich Netzwerkstatistiken, Client-Framerate und Ressourcennutzung.

Empfangen Sie Statistiken nach der Sitzung

Amazon GameLift Streams sammelt bei jeder Stream-Sitzung automatisch Leistungsstatistiken. Wenn Sie Sitzungsdateien exportieren, sind die Statistiken wie `stats/perf_stats_v1.csv` in der exportierten ZIP-Datei enthalten. Auf diese Weise erhalten Sie eine vollständige Aufzeichnung aller während der Sitzung gesammelten Statistiken für die Analyse und das Debugging nach der Sitzung.

Weitere Informationen zum Exportieren von Sitzungsdateien finden Sie unter [the section called “Exportieren Sie Stream-Sitzungsdateien”](#)

Referenz zu Leistungsstatistiken

In der folgenden Tabelle sind alle von Amazon GameLift Streams gesammelten Leistungsstatistiken aufgeführt. Anwendungsstatistiken sind spezifisch für die aktuelle Sitzung, wohingegen Statistiken gemeinsam genannter Systeme die Gesamtauslastung der gemeinsam genutzten Rechenleistung durch Sitzungen in Stream-Klassen mit mehreren Mandanten wiedergeben.

Normalisierte Statistiken zu Stream-Klassen mit mehreren Mandanten

Amazon GameLift Streams unterstützt Multi-Tenant-Stream-Klassen, bei denen sich mehrere Sessions dieselbe Recheninstanz teilen können. Normalisierte Statistiken (CPU- und Speicherauslastung der Anwendung) messen den Ressourcenverbrauch Ihrer Anwendung im Verhältnis zum zugewiesenen fairen Anteil. Der faire Anteil wird berechnet, indem die gesamte verfügbare CPU und der verfügbare Arbeitsspeicher auf der Recheninstanz gleichmäßig auf der Grundlage der Stream-Klassen-Tenancy aufgeteilt werden. Ein Wert von 1,0 bedeutet, dass Ihre Anwendung genau ihre Fair-Share-Allokation verwendet. Werte unter 1,0 bedeuten, dass Sie weniger als Ihre Zuteilung verwenden. Werte über 1,0 deuten auf eine Überlastung hin, was zu Leistungseinbußen bei Ihrer Sitzung führen kann. Bei Stream-Klassen mit mehreren Mandanten (Tenancy größer als 1) kann sich eine Überlastung auch auf andere Sitzungen auswirken, die dieselbe Recheninstanz gemeinsam nutzen.

Die in der folgenden Tabelle aufgeführten Statistiknamen werden in der exportierten Datei als CSV-Spaltenüberschriften verwendet. Wenn Sie Statistiken in Echtzeit über das Amazon GameLift Streams Web SDK erhalten, sind diese Statistiken über die `PerformanceStats` Schnittstelle mit Eigenschaftsnamen in Camel Case verfügbar. Die genaue Schnittstellenstruktur und die Eigenschaftsnamen finden Sie im Amazon GameLift Streams Web SDK API-Referenzhandbuch auf der [Produktseite Erste Schritte](#).

Statistikname (CSV-Spalte)	Description	Scope
<code>timestamp</code>	Uhrzeit, zu der die Messung durchgeführt wurde, im Format ISO 8601.	Alle
<code>app_cpu_normalized</code>	Die CPU-Auslastung der Anwendung wurde anhand der Fair-Share-Allokation normalisiert	Anwendung

Statistikname (CSV-Spalte)	Description	Scope
	ert, wobei 1,0 das angestrebte Fair-Share-Limit darstellt. Eine Auslastung über 1,0 weist auf eine Überauslastung hin, die zu Leistungseinbußen führen kann	
app_mem_mb	Gesamtspeicher (RAM), der von der Anwendung verwendet wird (gemessen in MiB)	Anwendung
app_mem_normalized	Die Speicherauslastung der Anwendung wurde anhand der Fair-Share-Allokation normalisiert, wobei 1,0 dem angestrebten Fair-Share-Limit entspricht. Eine Auslastung über 1,0 weist auf eine Überauslastung hin, die zu Leistungseinbußen führen kann	Anwendung
shared_sys_cpu_pct	Prozentsatz der gesamten CPU-Auslastung auf dem gemeinsam genutzten Computer.	Gemeinsames System
shared_sys_mem_mb	Gesamter auf der Instanz genutzter Speicher (gemessen in MiB).	Gemeinsames System
shared_sys_mem_pct	Prozentsatz des gesamten Arbeitsspeichers, der vom gemeinsam genutzten Computer genutzt wird.	Gemeinsames System
shared_sys_gpu_pct	Prozentsatz der gesamten GPU-Auslastung auf dem gemeinsam genutzten Computer.	Gemeinsames System
shared_sys_vram_mb	Gesamter VRAM (GPU-Speicher), der auf dem gemeinsam genutzten Computer verwendet wird (gemessen in MiB).	Gemeinsames System
shared_sys_vram_pct	Prozentsatz des gesamten VRAM (GPU-Speichers), der vom gemeinsam genutzten Computer verwendet wird.	Gemeinsames System

Problembehebung bei Amazon GameLift Streams

Themen

- [Zugriff verweigert, wenn eine Anfrage an den Amazon GameLift Streams-Service gestellt wurde](#)
- [Probleme mit der Anwendung](#)
- [Leistungsprobleme](#)
- [Probleme mit der Stream-Konnektivität und der Netzwerkleistung](#)
- [Probleme mit der Stream-Eingabe](#)
- [Probleme mit der Stream-Sitzung](#)
- [Kompatibilität mit Proton für Amazon GameLift Streams testen und beheben](#)
- [Profilierung der Leistung von Unreal Engine](#)

Zugriff verweigert, wenn eine Anfrage an den Amazon GameLift Streams-Service gestellt wurde

Wenn Sie bei Aufrufen von Amazon GameLift Streams APIs oder bei der Arbeit mit Ressourcen in der Konsole auf AccessDenied Ausnahmen stoßen, verfügt Ihre AWS Identity and Access Management (IAM-) Rolle möglicherweise nicht über ausreichende Berechtigungen für Amazon GameLift Streams. Überprüfen Sie, ob Folgendes der Fall ist:

- Wenn die IAM-Rolle eine explizite „Deny-All“ -Richtlinie hat, müssen Sie Amazon GameLift Streams explizit als Ausnahme von dieser Richtlinie auflisten, indem Sie das Element "gameliftstreams: *" hinzufügen. [NotAction](#) Beispiel:

```
{  
    "Sid": "DenyAllExceptListedIfNoMFA",  
    "Effect": "Deny",  
    "NotAction": [  
        "iam:CreateVirtualMFADevice",  
        "iam:EnableMFADevice",  
        "iam:GetUser",  
        "iam>ListMFADevices",  
        "iam>ListVirtualMFADevices",  
        "iam:ResyncMFADevice",  
        "sts:GetSessionToken",  
        "gameliftstreams:*" // Add this  
    ],  
    "Resource": "*",  
    "Condition": {  
        "BoolIfExists": {"aws:MultiFactorAuthPresent": "false"}  
    }  
}
```

- Weitere Informationen finden Sie [Identity and Access Management für Amazon GameLift Streams](#) im Kapitel Sicherheit und im IAM-Benutzerhandbuch unter [Behebung von Fehlermeldungen mit Zugriffsverweigerung](#).

Probleme mit der Anwendung

In diesem Abschnitt werden mögliche Ursachen für Probleme identifiziert, die verhindern, dass Anwendungen ausgeführt werden oder dass sie in Amazon GameLift Streams unterschiedlich angezeigt werden.

Vorläufige Prüfungen

- Führen Sie Ihre Anwendung auf einem anderen Computer aus, um zu überprüfen, ob sie korrekt verpackt ist. Dadurch wird bestätigt, dass Ihr Anwendungsinhalt keine fest codierten Pfade, fehlenden Ressourcen, Bibliotheken oder Binärdateien enthält, die auf anderen Geräten möglicherweise nicht funktionieren.

Probleme mit Proton

- Stellen Sie sicher, dass Ihre Anwendung mit Proton kompatibel ist. Testen Sie Ihre Anwendung in einer lokalen Umgebung ohne den Amazon GameLift Streams-Server, um sicherzustellen, dass sie mit Proton kompatibel ist. Detaillierte Anweisungen finden Sie unter [Kompatibilität mit Proton für Amazon GameLift Streams testen und beheben](#).

Anwendungsprobleme aufgrund der Bildschirmauflösung

Anwendungen können einfrieren, abstürzen oder falsch gerendert werden, wenn Sie versuchen, eine Vollbild-Auflösung zu verwenden, die nicht 1920x1080 ist. Wir empfehlen, für die Ausführung Ihrer Anwendung ein randloses Vollbild-Fenster zu verwenden und nicht zu versuchen, die Auflösung zu ändern.

Die Anwendung wird beim Start der Stream-Sitzung beendet

Wenn Ihre Anwendung sofort beendet wird, wenn eine Stream-Sitzung gestartet wird, überprüfen Sie die folgenden Punkte auf mögliche Ursachen und Lösungen:

- Überprüfen Sie die Laufzeit. Vergewissern Sie sich in der Amazon GameLift Streams-Anwendungskonfiguration, dass die Datei, die Sie im Startpfad der ausführbaren Datei angegeben haben, eine ausführbare Datei oder ein Skript ist und für die von Ihnen ausgewählte Laufzeitumgebung geeignet ist. Windows-Anwendungen sollten den Dateityp „.exe“, „.bat“ oder „.cmd“ haben und entweder auf die Windows- oder Proton-Laufzeiten abzielen. Native Linux-Anwendungen sollten ausführbare Dateien sein, die auf die Ubuntu 22.04 LTS-Laufzeit abzielen.
- Überprüfung erforderlich. DLLs Ihre Windows-Anwendung fehlt möglicherweise (erforderlich) DLLs. Wenn es sich bei Ihrer Anwendung beispielsweise um einen Debug-Build handelt, benötigt sie die Debug-Version der Microsoft C- und C++-Laufzeitbibliotheken (MSVC). Um dieses Problem zu lösen, empfehlen wir Ihnen, Ihren Build und zu verpacken. DLLs side-by-side Anweisungen finden Sie unter [Vorbereiten eines Testcomputers für die Ausführung einer ausführbaren Debug-Datei](#) von Microsoft.

Im Allgemeinen empfehlen wir, dass Sie Ihren Build zuerst auf einem sauberen Computer testen, bevor Sie es mit Amazon GameLift Streams versuchen. Anweisungen zum Testen auf einer EC2 Amazon-Instance finden Sie unter [Richten Sie einen Remote-Computer ein](#).

Die Unreal Engine-Anwendung stürzt ab oder erfordert zusätzliche Abhängigkeiten

Wenn Ihre Unreal Engine-Anwendung nicht gestartet werden kann, abstürzt oder Sie zusätzliche Abhängigkeiten wie die Microsoft C- und C++-Laufzeit (MSVC) installieren müssen, versuchen Sie Folgendes:

- Verwenden Sie die richtige ausführbare Datei. Damit Ihre Anwendung ordnungsgemäß mit Amazon GameLift Streams funktioniert, legen Sie den Anwendungspfad auf die vollständige ausführbare Datei fest, die sich im `Binaries/Win64/` (oder einem ähnlichen) Unterordner befindet. Unreal Engine erzeugt zwei ausführbare Dateien: eine kleine Bootstrap-Datei im Stammverzeichnis des Ordners und eine ausführbare Plattform-Zieldatei im Unterordner `Binaries/Win64/`. Die Bootstrap-Programmdatei im Stammverzeichnis versucht zu überprüfen, ob die Voreinstellungen korrekt sind, und kann in Amazon GameLift Streams zu Fehlalarmen führen, die den Anwendungsstart verhindern. Wenn die ausführbare Datei für das Plattformziel fehlt, wurde die Anwendung möglicherweise nicht korrekt erstellt. Sehen Sie sich zum Beispiel die folgende Ordnerstruktur einer Unreal-Beispielanwendung an:

```
BuildApp
| -> MyUnrealApp.exe
| -> MyUnrealApp
    | -> Binaries
        | -> Win64
            | -> MyUnrealApp.exe
```

- Schalten Sie Unreal Engine Assets aus. Deaktivieren Sie die `check`, `Makrosverify`, `undensure`. Sie können verhindern, dass die Anwendung Crash-Dumps erstellt. Weitere Informationen finden Sie in der Dokumentation zu [Assets in Unreal Engine](#).
 - Definieren Sie `USE_CHECKS_IN_SHIPPING=0` in Ihrem Build das Deaktivieren `check` und `verify` Makros.
 - Verwenden Sie das `-handleensurepercent=0` Befehlszeilenargument, um Makros zu deaktivieren. `ensure`

Leistungsprobleme

In diesem Abschnitt werden mögliche Ursachen für Leistungsprobleme bei der Ausführung von Spielen auf Amazon GameLift Streams identifiziert und Vorschläge zur Optimierung Ihrer Streams im Service gegeben.

Die Spielleistung wird reduziert, wenn auf Amazon GameLift Streams gestreamt wird

Wenn Ihr Spiel auf Ihrem eigenen Computer gut läuft, aber Leistungsprobleme auftreten, wenn Sie es auf Amazon GameLift Streams streamen, sollten Sie Folgendes beachten:

- Ihr Computer verfügt möglicherweise über eine leistungsfähigere Hardware als Amazon GameLift Streams. Stellen Sie sicher, dass Sie die Anwendung auf einem Computer mit ähnlicher Leistung wie die von Amazon GameLift Streams verwendete Hardware testen:
 - gen4n: vergleichbar mit der NVIDIA RTX 2060 GPU
 - gen5n: vergleichbar mit der NVIDIA RTX 3080 GPU
 - gen6n: vergleichbar mit der NVIDIA RTX 4060 GPU

Dadurch wird überprüft, ob die Rendering-Einstellungen Ihrer Anwendung mit der GPU kompatibel sind und ob die Leistung Ihren Erwartungen entspricht.

- Das Problem könnte an Ihrer Netzwerkverbindung oder den Einstellungen von Amazon GameLift Streams liegen. Probieren Sie die Tipps zur Fehlerbehebung im [Probleme mit der Stream-Konnektivität](#) Abschnitt aus.

Wenn dein Spiel langsam ist, auch wenn es lokal läuft, musst du die Leistung optimieren. Die besten Optimierungsmethoden hängen von der spezifischen Engine oder dem Framework ab, das du verwendest.

- Unreal Engine-Spiele finden Sie unter. [Profilierung der Leistung von Unreal Engine](#)

Bei Windows-Anwendungen treten langsame Ladezeiten oder ruckelnde Probleme auf

Wenn Ihr Spiel lange Ladezeiten hat oder stottert, empfehlen wir die folgende Vorgehensweise:

1. Stellen Sie sicher, dass Ihre Anwendung paketiert und für die Ladeleistung optimiert ist. Beachten Sie dabei die Hinweise Ihres Engine-Anbieters zur Optimierung von Inhalt und Shader-Leistung.
2. Stellen Sie sicher, dass Ihre Anwendung als [Standardanwendung](#) in einer Stream-Gruppe festgelegt ist.
3. Optimieren Sie den ersten Start der Anwendung im Service, indem Sie Shader als Teil Ihrer Anwendungspaketierung zwischenspeichern.

Es gibt zwei Ansätze, um das Shader-Caching zu aktivieren:

- Treiberbasiertes Caching — Dieser Ansatz ist spezifisch für die GPU- und Treiberversion der Laufzeitumgebung. Diese Option kann auf alle Anwendungen angewendet werden und ist daher der empfohlene Standardansatz. Die Schritte für diesen Ansatz müssen für jede GPU/driver Kombination repliziert werden.
- Engine-basiertes Caching — Dieser Ansatz ermöglicht Shader-Caching über die Game-Engine, sofern verfügbar. Dadurch wird der Entwickler mit der Erstellung eines vorgefertigten Pipeline-State-Object-Caches (PSO) betraut. Außerdem wird davon ausgegangen, dass die Engine in der Lage ist, die Cache-Unterstützung für verschiedene Treiber auf derselben GPU-Hardware zu verarbeiten.

Als bewährte Methode empfehlen wir, zuerst treiberbasiertes Caching zu implementieren, da hierfür kein tiefes Verständnis der Implementierung von PSO-Caching für die jeweilige Engine erforderlich ist.

Mit diesen Implementierungen können Shaderdateien exportiert und zusammen mit Ihrer Anwendung gepackt werden, sodass sie nicht bei jedem neuen Stream-Start generiert werden müssen.

Um einen treiberbasierten Caching-Fix für eine Windows-Runtime-Anwendung zu implementieren

1. Starten Sie das Streaming Ihrer Standardanwendung und spielen Sie sie ausgiebig ab, um Shader für die Anwendung zu generieren.

 **Important**

Achten Sie darauf, alle Bereiche oder Ebenen der Umgebung zu besuchen, um so viele Shader wie möglich zu erzeugen.

2. Bevor Sie den Stream schließen, aktivieren Sie die Exportfunktion in Ihrer aktiven Stream-Sitzung. Details hierzu finden Sie unter [Stream-Sitzungsdateien exportieren](#).
3. Laden Sie die .zip-Datei für den Stream-Sitzungsexport aus dem Amazon S3 S3-Bucket herunter, den Sie im vorherigen Schritt angegeben haben. Einen Download-Link finden Sie in der Amazon GameLift Streams-Konsole auf der Seite Sessions.
4. Suchen Sie den Shader-Ordner innerhalb des Stream-Session-Exports. Es wird normalerweise an diesem Ort gespeichert:AppData\Local\NVIDIA\DXCache. Laden Sie die generierten Shader-Dateien (*.nvph) in den Amazon S3 S3-Bucket Ihrer Anwendung hoch.
5. Erstellen Sie eine .bat Datei, die die Shader-Dateien zur Laufzeit in den NVIDIA-Caching-Ordner kopiert. Dieser Ordner befindet sich normalerweise unter: C:\Users\Administrator\AppData\Local\NVIDIA\DXCache. Laden Sie die .bat Datei in den Amazon S3 S3-Anwendungs-Bucket hoch.
6. Erstellen Sie eine neue Amazon GameLift Streams-Anwendung mit der .bat Datei als ausführbarem Pfad.

Wenn Ihre Anwendung mit dem Streaming beginnt, kopiert Ihre .bat Datei die vorgenerierten Shader in den Shader-Cache, bevor die Anwendung gestartet wird, wodurch die Leistung beim Laden des Streams verbessert wird.

 Note

Möglicherweise müssen Sie diese Schritte wiederholen, wenn Sie Ihre Anwendung aktualisieren oder die Amazon GameLift Streams-Anwendung mit einer neuen Stream-Gruppe verknüpfen. Neuere Stream-Gruppen können aktualisierte GPU-Treiber aus dem Service enthalten.

In der folgenden .bat Beispieldatei wird davon ausgegangen, dass die Shader-Dateien unter dem Amazon S3 S3-Bucket-Präfix Shaders\ gespeichert sind. Sie können eine andere Ordnerstruktur verwenden.

```
@echo off
set CURRENT_PATH=%cd%
set DXCACHE_DIR=%CURRENT_PATH%\Shaders
set NVIDIA_DXCACHE_DIR=C:\Users\Administrator\AppData\Local\NVIDIA\DXCache

if not exist "%NVIDIA_DXCACHE_DIR%" (
```

```
mkdir "%NVIDIA_DXCACHE_DIR%"  
)  
  
xcopy /s /f "%DXCACHE_DIR%" "%NVIDIA_DXCACHE_DIR%"  
  
start %CURRENT_PATH%\app.exe
```

Um einen treiberbasierten Caching-Fix für eine Proton-Runtime-Anwendung zu implementieren

1. Starten Sie das Streaming Ihrer Standardanwendung mit der folgenden Umgebungsvariablenüberschreibung:

```
"__GL_SHADER_DISK_CACHE_PATH" : "/home/unpriv/games"
```

2. Spielen Sie die Anwendung ausgiebig ab, um Shader zu generieren.

⚠ Important

Achten Sie darauf, alle Bereiche oder Ebenen der Umgebung zu besuchen, um so viele Shader wie möglich zu erzeugen.

3. Bevor Sie den Stream schließen, aktivieren Sie die Exportfunktion in Ihrer aktiven Stream-Sitzung. Details hierzu finden Sie unter [Stream-Sitzungsdateien exportieren](#).
4. Laden Sie die .zip-Datei für den Stream-Sitzungsexport aus dem Amazon S3 S3-Bucket herunter, den Sie im vorherigen Schritt angegeben haben. Einen Download-Link finden Sie in der Amazon GameLift Streams-Konsole auf der Seite Sessions.
5. Suchen Sie die Shader-Ordner und -Dateien innerhalb des Stream-Session-Exports:
 - a. Ordner application\GLCache
 - b. wenn die Anwendung DX11: application\path-to-exe\exe-name.dxvk-cache Datei verwendet
 - c. wenn die Anwendung verwendet DX12: application\path-to-exe\vkd3d-proton.cache.write Datei
6. Laden Sie die generierten Shader-Dateien in den Amazon S3 S3-Bucket Ihrer Anwendung hoch:
 - a. Kopieren Sie den GLCache Ordner in das Stammverzeichnis Ihrer Anwendung.

- b. Falls verfügbar, kopieren Sie die Datei `.dxvk-cache` oder die `vkd3d-proton.cache.write` Cache-Datei in den Ordner, der die ausführbare Datei der Anwendung enthält.
7. Erstellen Sie eine neue Amazon GameLift Streams-Anwendung mit derselben Proton-Konfiguration.
8. Führen Sie die Anwendung mit derselben Umgebungsvariablenüberschreibung aus:

```
"__GL_SHADER_DISK_CACHE_PATH" : "/home/unpriv/games"
```

Wenn Ihre Anwendung mit dem Streaming beginnt, verwendet sie die vorgenerierten Shader, wodurch die Leistung beim Laden des Streams verbessert wird.

 Note

Möglicherweise müssen Sie diese Schritte wiederholen, wenn Sie Ihre Anwendung aktualisieren oder die Amazon GameLift Streams-Anwendung mit einer neuen Stream-Gruppe verknüpfen. Neuere Stream-Gruppen können aktualisierte GPU-Treiber aus dem Service enthalten.

Um einen Engine-basierten Caching-Fix für eine Anwendung zu implementieren, die Unreal Engine verwendet

Für diesen Ansatz können Sie die Funktionen von Unreal Engine verwenden, um einen Pipeline State Object (PSO) -Cache für Ihre Amazon GameLift Streams-Anwendung zu erstellen. Mit einem PSO-Cache können Sie vorkomplizierte Grafik-Pipeline-Status mit kürzeren Kompilierungszeiten zur Laufzeit bereitstellen, wodurch Probleme beim Laden und Rendern reduziert werden können. Dies erfordert fortgeschrittene Kenntnisse der Unreal Engine, weshalb wir hier nicht auf alle Engine-spezifischen Details eingehen werden. Weitere Anweisungen finden Sie in der Anleitung von Unreal Engine im Abschnitt „Sammlungsablauf“ unter „[Einen gebündelten PSO-Cache erstellen](#)“.

1. Generieren Sie Shader für Ihre Anwendung, für die die PSO-Protokollierung aktiviert ist.
 - a. Erstellen Sie eine neue Amazon GameLift Streams-Anwendung mithilfe des Build-Pakets mit der PSO-fähigen Anwendung.

- b. Starten Sie einen Stream mit einem -logPSO Befehl in Ihrer PSO-Logging-App. Sie können die Option Befehlszeilenargumente auf der Seite Stream-Konfiguration testen in der Amazon GameLift Streams-Konsole verwenden.

 **Important**

Achten Sie darauf, alle Bereiche oder Ebenen der Umgebung zu besuchen, um so viele Shader wie möglich zu erzeugen.

- c. Bevor Sie den Stream schließen, aktivieren Sie die Exportfunktion in Ihrer aktiven Stream-Sitzung. Details hierzu finden Sie unter [Stream-Sitzungsdateien exportieren](#).
 - d. Beenden Sie die Anwendung über das Menü oder mithilfe der Unreal-Shutdown-Befehle. Wenn Sie den Stream direkt schließen, wird die Unreal-Shader-Pipeline-Datei nicht generiert.
 - e. Laden Sie die .zip-Datei für den Stream-Sitzungsexport aus dem Amazon S3 S3-Bucket herunter, den Sie im Exportschritt angegeben haben. Einen Download-Link finden Sie in der Amazon GameLift Streams-Konsole auf der Seite Sessions.
2. Package Sie die Unreal-Shader-Pipeline-Datei in Ihre Amazon GameLift Streams-Anwendung.
 - a. Suchen Sie die aufgezeichneten PSO-Dateien (rec.pipelinecache) im Stream-Sitzungsexport unter Saved/CollectedPSOs. Entpacken Sie die PSO-Dateien mithilfe von Unreal-Befehlen.
 - b. Packen Sie einen neuen Unreal-Build mit der generierten Ausgabe aus dem Entpacken. Folgen Sie den Anleitungen von Unreal in den Abschnitten PSO-Caches [konvertieren und PSO-Caches](#) in Ihre Anwendung [einbeziehen](#).

 **Important**

Wenn Sie den Unreal-Befehl im Abschnitt „PSO-Caches konvertieren“ ausführen, stellen Sie sicher, dass Sie die Eingabedateien derselben Treiberversion verwenden. Beispiel: Verwenden Sie für DX12 nur die SM6 Dateien als Eingaben. Andernfalls erhalten Sie beim Verpacken der neuen Anwendung eine Fehlermeldung.

- c. Erstellen Sie eine neue Amazon GameLift Streams-Anwendung für den neuen Paket-Build mit den PSO-Dateien.

- d. Stellen Sie beim Starten und Testen von Streams sicher, dass der PSO-Cache geladen wird. Suchen Sie in den Spielprotokollen nach der folgenden Zeile:

```
Opened FPipelineCacheFile: ../../...
```

 Note

Möglicherweise müssen Sie diese Schritte wiederholen, wenn Sie Ihre Anwendung aktualisieren oder die Amazon GameLift Streams-Anwendung mit einer neuen Stream-Gruppe verknüpfen. Neuere Stream-Gruppen können aktualisierte GPU-Treiber aus dem Service enthalten.

Probleme mit der Stream-Konnektivität und der Netzwerkleistung

Wenn Sie [Ihren Amazon GameLift Streams-Backend-Service einrichten](#), überprüfen Sie Folgendes:

- Wählen Sie eine Lösung, AWS-Region die dem Endbenutzer am nächsten kommt. Eine hohe Latenz zwischen Ihren Kunden und der Region, in der Ihr Stream gehostet wird, kann sich auf die Stream-Qualität auswirken. Eine Liste der Standorte, [AWS-Regionen und entfernte Standorte, die von Amazon GameLift Streams unterstützt werden](#) von denen aus Sie streamen können, finden Sie unter. Sie können AWS Konsolenendpunkte in der Region anpingen, um eine ungefähre Latenzmessung zu erhalten.
- Stellen Sie sicher, dass Ihre Stream-Gruppe Kapazität für neue Streams hat.
- Vergewissern Sie ConnectionTimeoutSeconds sich, dass diese Einstellung so gewählt ist, dass Endbenutzer genügend Zeit haben, eine Verbindung herzustellen, bevor ihr Webclient das Timeout erreicht.

Empfehlen Sie Ihren Endbenutzern, Folgendes zu überprüfen:

- Stellen Sie sicher, dass Firewalls den Zugriff auf den UDP-Portbereich 33435-33465 zulassen, um Streaming von Amazon Streams zu ermöglichen. GameLift Wenn Amazon GameLift Streams diese Ports nicht erreichen kann, kann dies zu Streaming-Problemen führen, z. B. zu einem schwarzen oder grauen Bildschirm.

- Stellen Sie sicher, dass Ihre Internetverbindung eine Verbindungsgeschwindigkeit von mindestens 10 Mbit/s für einen 1080p-Stream aufrechterhalten kann. Wenn Sie beim Abspielen auf Amazon GameLift Streams Netzwerkprobleme feststellen, schwankt Ihre Internetgeschwindigkeit möglicherweise und Sie erhalten möglicherweise nicht durchgehend mindestens 10 Mbit/s. Führen Sie einen Internet-Geschwindigkeitstest durch und fahren Sie mit den Schritten zur Fehlerbehebung fort.
- Verwenden Sie nach Möglichkeit ein verkabeltes Netzwerk. Wenn Sie WLAN verwenden, stellen Sie Ihr Gerät in die Nähe Ihres Routers, um eine stärkere Signalstärke zu erzielen.
- Wenn Sie einen Wi-Fi-Router mit 2,4- GHz und GHz 5-Bändern verwenden, versuchen Sie, eine Verbindung zu einem anderen Band herzustellen. Wenn Sie sich nicht sicher sind, wie Sie Ihren Router auf ein anderes Band umschalten sollen, besuchen Sie die Support-Seiten des Herstellers oder Anbieters Ihres Wi-Fi-Routers. Sie können sich auch an den Kundendienst wenden.
- Stellen Sie fest, ob andere Benutzer im selben Netzwerk (insbesondere im WLAN zu Hause) Anwendungen mit hoher Bandbreite wie Videostreaming, Herunterladen, Online-Spiele oder Backups ausführen.
- Schließen Sie andere Anwendungen auf Ihrem Gerät, die Bandbreite beanspruchen.
- Verwenden Sie beim Streamen kein VPN oder Proxy. Sie können zu höheren Latenzen führen und das Gameplay beeinträchtigen.
- Vergewissere dich, dass du WLAN anstelle von Mobilfunknetzen verwendest, wenn du auf einem iPad oder iPhone spielst. Die Verwendung eines Mobilfunknetzes kann zu Verbindungsproblemen führen.
- MacOS-Benutzer sollten die Ortungsdienste deaktivieren, da dadurch das WLAN von Zeit zu Zeit unterbrochen wird, was zu einem schlechten Streaming-Erlebnis führt.

Probleme mit der Stream-Eingabe

In diesem Abschnitt werden mögliche Ursachen und Lösungen für Probleme im Zusammenhang mit Benutzereingaben in einer Stream-Sitzung beschrieben.

Allgemeine Problembehebung bei Eingaben

- Testen Sie, ob das Problem browserspezifisch ist. Insgesamt empfehlen wir Google Chrome, Microsoft Edge oder eine benutzerdefinierte Chromium-basierte Desktop-Anwendung für das beste Endbenutzererlebnis und maximale Kompatibilität, insbesondere mit Gamecontrollern.

- Protokollieren Sie Eingabeereignisse, die vom Client gesendet und von der Anwendung empfangen wurden, um festzustellen, wo ein Eingabefehler in Ihrem Frontend-Code vorliegt.
- Suchen Sie unbedingt [Unterstützte Browser und Eingaben](#) nach zusätzlichen Informationen zu unterstützten Browsern und Eingabegeräten, einschließlich bekannter Probleme und Einschränkungen.

Gamepad- und Mikrofoneingänge funktionieren in nativen Linux-Anwendungen nicht

Gamepad- und Mikrofoneingänge werden in nativen Linux-Anwendungen nicht unterstützt.

[Unterstützte Browser und Eingaben](#) Weitere Informationen zu unterstützten Eingabegeräten, einschließlich bekannter Probleme und Einschränkungen, finden Sie unter.

Die Tasteneingabe scheint auf dem macOS-Client hängen zu bleiben

Auf macOS-Clients scheinen Tasten plötzlich festzustecken, wenn die Befehlstaste und eine andere Taste gleichzeitig gedrückt werden, wodurch das Tastenereignis wiederholt wird. Beispielsweise kann die Pfeiltaste hängen bleiben, wenn auch die Befehlstaste gedrückt wird. Wenn in einem Spiel die Pfeiltasten verwendet werden, um die Kamera zu drehen, würde dies dazu führen, dass sich die Kamera endlos dreht.

- Problem: Die Befehlstaste auf macOS ist dem Metastaten-Ereignis zugeordnet, das der Windows-Taste unter Microsoft Windows zugeordnet ist. Das Problem ist ein [Fehler](#), der sich auf macOS-Browser auswirkt, wenn Command und eine andere Taste gleichzeitig gedrückt werden. Dabei wird die Metastaste zurückgesetzt, wenn sie losgelassen wird, aber die Pfeiltaste nicht zurückgesetzt, weil der Browser kein Keyup-Ereignis für die Pfeiltaste erfasst hat, sodass der Web SDK-Client kein Keyup-Ereignis an den Server sendet und die Streaming-Anwendung immer noch denkt, dass die Taste gedrückt wird.
- Lösung: Wenn Sie die Befehlstaste nicht verwenden, können Sie sie mithilfe des Web-SDK-Tastaturfiltermechanismus (`keyboardFilter`) herausfiltern, der sich in der `InputConfiguration` Benutzeroberfläche des Web SDK befindet.

Mausbewegungen verhalten sich in Amazon GameLift Streams anders

Wenn sich Mausbewegungen beim Streamen mit Amazon GameLift Streams anders verhalten, z. B. schneller als erwartet, müssen Sie möglicherweise die Maussteuerung und die Cursorverwaltungslogik in Ihrer Anwendung anpassen.

- Problem: Amazon GameLift Streams verwendet eine Heuristik, um auszuwählen, ob Mausereignisse im „relativen“ oder „absoluten“ Modus übertragen werden sollen. Im relativen Modus werden neue Maus-Updates als kleine, inkrementelle Unterschiede zum vorherigen Update bereitgestellt. Im absoluten Modus wird der Mauszeiger kontinuierlich an eine Bildschirmposition gedrückt, die mit dem Client synchronisiert ist. Wenn der Betriebssystem-Cursor über dem gestreamten Inhalt sichtbar ist, wählt die Heuristik immer absolute Koordinaten aus. Dies kann zu unerwartet großen Bewegungsdeltas führen, wenn Ihre Anwendung kleine, relative Aktualisierungen erwartet.
- Lösung: Wenn Ihre Anwendung relative Mausbewegungen erwartet (z. B. Kamerasteuerung im FPS-Stil oder Interaktionen auf Ziehbasis), blenden Sie den Betriebssystem-Cursor bei Mausinteraktionen aus. Sie können den Cursor beispielsweise bei gedrückter Maus ausblenden und bei gedrückter Maus wieder einblenden. Dadurch wird sichergestellt, dass bei Ziehbewegungen relative Koordinaten verwendet werden und die absolute Position nur synchronisiert wird, wenn die Taste losgelassen wird.

Weitere Informationen zur Mausbewegung in Amazon GameLift Streams finden Sie unter [Umgang mit Mausbewegungen](#).

Probleme mit der Stream-Sitzung

In diesem Abschnitt werden mögliche Ursachen und Lösungen für Probleme im Zusammenhang mit dem unerwarteten Starten oder Beenden einer Stream-Sitzung beschrieben.

Die Stream-Sitzung wird nicht gestartet

Mögliche Ursachen:

- Die Anwendung hängt oder ist abgestürzt. Anweisungen zur Fehlerbehebung finden Sie [Probleme mit der Anwendung](#) im Abschnitt.
- Der Status der Stream-Gruppe ist `nichtActive`. Überprüfen Sie den Status der Stream-Gruppe.

- Das Hochfahren der On-Demand-Kapazität dauert länger als das ConnectionTimeoutSeconds in der [StartStreamSession](#)API angegebene Timeout. In der Windows-Runtime kann die On-Demand-Hochlaufzeit 5 Minuten oder länger dauern.
- Am Streaming-Standort ist keine Kapazität verfügbar. Stellen Sie sicher, dass Ihre zugewiesene Kapazität größer ist als Ihre genutzte Kapazität oder ob Sie über On-Demand-Kapazität verfügen, die nicht genutzt wird (die zugewiesene Kapazität ist geringer als die ständig verfügbare Kapazität plus On-Demand-Kapazität). In der Konsole finden Sie diese Werte in der Liste der Stream-Gruppen oder auf der Detailseite der Stream-Gruppe. Mithilfe der Service-API finden Sie diese Werte unter [GetStreamGroup](#). Zu den Szenarien, in denen die verfügbare Kapazität vorübergehend bei Null liegt, gehören die folgenden:
 - Wenn Sie gerade die Always-On-Kapazität am Streaming-Standort erhöht haben, warten Sie einige Minuten, bis die Kapazität zugewiesen ist.
 - Wenn am Streaming-Standort nur eine Kapazität verfügbar ist und Ihr Client die Verbindung unerwartet unterbrochen hat, befindet sich die vorherige Sitzung möglicherweise immer noch im Status „Nicht verbunden“. Warten Sie einige Minuten, bis das Timeout für die Sitzung abgelaufen ist, und versuchen Sie es erneut.
 - Wenn Sie Ihrer Stream-Gruppe kürzlich einen Standort hinzugefügt haben und die Anwendung an diesem Standort nicht vorhanden war, hat die Anwendung die Replikation dort möglicherweise noch nicht abgeschlossen. Überprüfen Sie den Replikationsstatus auf der Seite mit den Stream-Gruppendetails in der Konsole. Alternativ können Sie die [GetApplication](#)API verwenden und den ReplicationStatuses Wert überprüfen, um zu überprüfen, ob es sich um den gewünschten Streaming-Standort handeltCOMPLETED. Status
- Die Netzwerkbedingungen sind so schlecht, dass Frames, insbesondere der erste Frame, nicht gesendet werden. Überprüfen Sie die Netzwerkbedingungen zwischen dem Client und dem Streaming-Standort und passen Sie einen anderen Standort an oder versuchen Sie es mit einem anderen.

Die Stream-Sitzung wurde beendet

Stream-Sitzungen werden automatisch beendet, wenn eine Anwendung abstürzt oder beendet wird oder wenn die Client-Verbindung unterbrochen wird. Sitzungen können auch aufgrund der folgenden Timeout-Werte beendet werden:

- Platzierungs-Timeout: Timeout-Wert für Amazon GameLift Streams, um Rechenressourcen für das Hosten einer Stream-Sitzung zu finden.

- Verbindungs-Timeout: Timeout-Wert für einen Client, um sich mit einer Stream-Sitzung zu verbinden oder erneut zu verbinden.
- Timeout im Leerlauf: Maximale Zeit, während der eine Stream-Sitzung ohne Benutzereingabe inaktiv sein kann.
- Timeout für die Sitzungslänge: Maximale Zeit für eine Stream-Sitzung.

Eine ausführliche Erläuterung der einzelnen Timeouts und ihrer möglichen Werte finden Sie unter.

[Timeout-Werte, die sich auf Stream-Sitzungen auswirken](#)

Kompatibilität mit Proton für Amazon GameLift Streams testen und beheben

Wenn Ihre Amazon GameLift Streams-Anwendung in einer Proton-Laufzeitumgebung ausgeführt wird, kann Ihnen dieser Abschnitt bei der Behebung von Kompatibilitätsproblemen zwischen Ihrer Anwendung und der Proton-Schicht helfen. Diese Anweisungen enthalten eine Reihe von Skripten, die Proton auf Ihrem eigenen Computer installieren und die Umgebung simulieren, die Amazon GameLift Streams verwenden würde. Durch die Fehlerbehebung ohne den Amazon GameLift Streams-Service können Sie sich auf die Behebung von Problemen konzentrieren, die für Ihre Anwendung und die Laufzeitumgebung spezifisch sind.

Allgemeine Schritte zum Testen und zur Fehlerbehebung

1. Erwerben Sie einen Ubuntu 22.04 LTS-Computer. Sie können entweder einen lokalen Computer oder einen EC2 Cloud-basierten Amazon-Desktop verwenden. Anleitungen finden Sie unter den folgenden Themen:
 - [Richten Sie einen lokalen Computer ein](#)
 - [Richten Sie einen Remote-Computer ein](#)
2. Installieren Sie die Proton-Laufzeitumgebung, um Ihre Anwendung zu testen und zu debuggen. Anleitungen finden Sie [Fehlerbehebung bei Proton](#) unter.

Bekannte Probleme mit Proton

Die neuesten Ressourcen zur Kompatibilität und Fehlerbehebung finden Sie im [GitHub Proton-Wiki](#). Sie können auch im Proton GitHub [Issue Tracker](#) nach Problemen suchen. Im Folgenden

sind einige spezifische Probleme aufgeführt, auf die unsere Kunden beim Ausführen von Windows-Anwendungen auf Proton gestoßen sind:

Godot-Anwendungen auf Proton

- Godot-basierte Anwendungen, die auf Proton ausgeführt werden, können auf einen schwarzen Bildschirm stoßen, wenn die Amazon Vulkan-Capture-Layer aktiviert ist. Um dieses Problem zu beheben, deaktivieren Sie beim Streaming gemeinsam genutzte Texturen, indem Sie die Umgebungsvariable festlegen. `VK_LAYER_AMZN_BLITSURFACE_SHARED_TEXTURES=0`

Unreal Engine-Anwendungen auf Proton

- [Wenn Sie Probleme mit Proton 8.x mit Electra Media Player \(einem Unreal Engine-Plugin\) haben, empfehlen wir, die in wine/pull/257 enthaltenen Korrekturen zu verwenden. <https://github.com/ValveSoftware/>](#)

Richten Sie einen lokalen Computer zur Fehlerbehebung bei Proton ein

Proton ist eine Kompatibilitätsebene, mit der Windows-Anwendungen unter Linux ausgeführt werden können. Daher benötigen Sie einen Ubuntu-Computer, mit dem Sie testen und Fehler beheben können. Wenn Sie keinen lokalen Ubuntu-Computer haben, können Sie mit Amazon einen Remote-Computer einrichten EC2. Folgen Sie dazu [Richten Sie einen Remote-Computer ein](#) stattdessen den Schritten unter.

Voraussetzungen

- [Ubuntu 22.04 LTS](#). Für Installationsanweisungen können Sie das Ubuntu-Tutorial [Install Ubuntu Desktop](#) verwenden.
- NVIDIA-GPU

Installieren Sie GPU-Treiber

Durch die Installation der neuesten GPU-Treiber kann eine schlechte Leistung Ihrer Anwendung und Abstürze verhindert werden.

Um zu überprüfen, welchen GPU-Treiber Ihr System verwendet

1. Führen Sie den folgenden Befehl von einem Terminal aus:

```
lshw -C display | grep driver
```

2. Wenn die richtigen Treiber installiert sind, sollten Sie die folgende Ausgabe oder eine ähnliche Ausgabe sehen, wo <gpu> ist nvidia für NVIDIA: configuration: driver=<gpu> latency=0

Um die neuesten NVIDIA-GPU-Treiber zu installieren

Folgen Sie den Anweisungen unter [Installation der NVIDIA-Treiber](#).

Überprüfen Sie die GPU-Treiber

Stellen Sie sicher, dass die GPU-Treiber installiert sind und ordnungsgemäß funktionieren. Eine Möglichkeit, dies zu überprüfen, besteht darin, die [vkcube-Anwendung](#) in einem Terminal auszuführen.

1. Installieren Sie das vulkan-tools apt-Paket mit dem folgenden Befehl.

```
sudo apt install -y vulkan-tools
```

2. Führen Sie vkcube.

3. Überprüfen Sie die Ausgabe.

- Wenn Ihr System die richtige GPU ordnungsgemäß verwendet, erhalten Sie eine Ausgabe, die der folgenden ähnelt, mit dem Namen Ihrer GPU: Selected GPU 0: AMD Radeon Pro V520 (RADV NAVI12), type: 2
- Wenn Ihre Anwendung die GPU nicht korrekt verwenden kann, wird möglicherweise eine andere Ausgabe angezeigt, die der folgenden ähnelt: Selected GPU 0: llvmpipe (LLVM 15.0.7, 256 bits), type: 4

Überprüfen Sie in diesem Fall die GPU-Treiber und installieren Sie sie bei Bedarf erneut.

Nächster Schritt

Wenn Ihr lokaler Ubuntu-Computer bereit ist, besteht der nächste Schritt darin, Proton einzurichten. Anweisungen finden Sie unter [Fehlerbehebung bei Proton](#)

Richten Sie einen EC2 Remote-Amazon-Computer zur Fehlerbehebung bei Proton ein

Wenn Sie keinen lokalen Ubuntu-Computer haben, folgen Sie diesen Anweisungen, um stattdessen einen Remote-Computer einzurichten.

In diesem Schritt richten Sie Ihren Remote-Ubuntu-Computer mithilfe von Amazon Elastic Compute Cloud (Amazon EC2) ein, mit der Sie die Kompatibilität Ihrer Anwendung mit Proton für Amazon GameLift Streams beheben können. In diesem Thema wird beschrieben, wie Sie eine EC2 Amazon-Instance mit Ubuntu 22.04 LTS, den erforderlichen GPU-Treibern und dem Amazon DCV-Server für einen visuellen Remote-Desktop einrichten.

Starten Sie eine EC2 Amazon-Instance mit Ubuntu 22.04 LTS AMI

1. Navigieren Sie EC2 in der AWS-Managementkonsole zu Amazon.
2. Wählen Sie Launch Instances aus.
3. Geben Sie „Amazon GameLift Streams Testing“ als Namen ein.
4. Wählen Sie Ubuntu Server 22.04 LTS (HVM) für Anwendungs- und Betriebssystem-Images (Amazon Machine Image).
5. Wählen Sie g4dn.2xlarge als Instance-Typ aus.
6. Wählen Sie für key pair (Anmeldung) ein Schlüsselpaar aus, wenn Sie SSH für den Zugriff auf die Instance verwenden möchten. Wir empfehlen, ein Instanzprofil mit der `AmazonSSMManagedInstanceCore` Richtlinie zu verwenden, mit der Sie eine Verbindung zu Ihren Instances herstellen AWS Systems Manager Session Manager. Weitere Informationen finden Sie unter [Hinzufügen von Session Manager-Berechtigungen zu einer vorhandenen IAM-Rolle](#).
7. Erstellen Sie für Netzwerkeinstellungen eine neue Sicherheitsgruppe:
8. Geben Sie DCV als Namen der Sicherheitsgruppe ein.
9. Fügen Sie Regeln für eingehende Sicherheitsgruppen mit TypCustom TCP, Portbereich und Quelltyp `0.0.0.0/0`, Anywhere um den Zugriff über Amazon DCV zu ermöglichen.
10. Erhöhen Sie den Speicher auf mindestens 256 GB und wählen Sie gp3 als Speichertyp.
11. Wählen Sie Launch Instance aus.

Ihre Instance sollte jetzt gestartet werden.

Folgen Sie den Anweisungen unter [Connect zu Ihrer Linux-Instance](#) herstellen, um sich mit der Instance über SSH oder AWS Systems Manager Session Manager zu verbinden.

Installieren Sie die GPU-Treiber

G4dn - NVIDIA-GPU

Installieren Sie zusätzliche Module und Linux-Firmware, indem Sie die folgenden Befehle ausführen:

```
sudo apt install linux-modules-extra-aws linux-firmware

# Install the AWS CLI required for NVIDIA driver installation
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
sudo apt install unzip
unzip awscliv2.zip
sudo ./aws/install
```

Folgen Sie den Anweisungen zu den NVIDIA GRID-Treibern für Ubuntu und Debian unter [NVIDIA-Treiber unter Linux installieren](#).

Richten Sie die Benutzerumgebung ein

Richten Sie Ihre Benutzerumgebung so ein, dass sie die GPU verwenden kann, indem Sie die folgenden Befehle ausführen. Dies macht die folgenden Dinge:

- Fügen Sie Sie den `video` Gruppen hinzu, um Zugriff auf ein Videogerät zu erhalten, und der `render` Gruppe, um Ihnen Zugriff auf ein Rendergerät zu gewähren.
- Installieren Sie den AWS CLI, der für NVIDIA-Treiber und zum Herunterladen Ihrer Anwendungen oder Spiele von Amazon S3 erforderlich ist.

```
sudo adduser user

# Add the current user to the video and render group
sudo usermod -a -G video user
sudo usermod -a -G render user
sudo adduser user sudo

# Install the AWS CLI
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
sudo apt install unzip
unzip awscliv2.zip
sudo ./aws/install

sudo reboot
```

Installation und Konfiguration von Amazon DCV

Stellen Sie mithilfe von SSH oder über SSH erneut eine Verbindung zur Instance her AWS Systems Manager Session Manager und folgen Sie den Anweisungen unter [Installation des Amazon DCV-Servers unter Linux für Ubuntu](#).

- Stellen Sie sicher, dass der Server korrekt konfiguriert ist, wie in der Dokumentation beschrieben.
- Folgen Sie den Schritten unter [Installieren und Konfigurieren von NVIDIA-Treibern](#) für NVIDIA-GPU.
- Fügen Sie den Amazon DCV-Benutzer zur Videogruppe hinzu, wie in [Schritt 7 des Handbuchs Installation des Servers](#) erklärt (navigieren Sie zur Registerkarte Ubuntu).

Es ist nicht erforderlich, optionale Teile des Amazon DCV-Servers zu installieren.

Wenn Sie fertig sind, führen Sie den folgenden Befehl aus, um den Amazon DCV-Server zu starten:

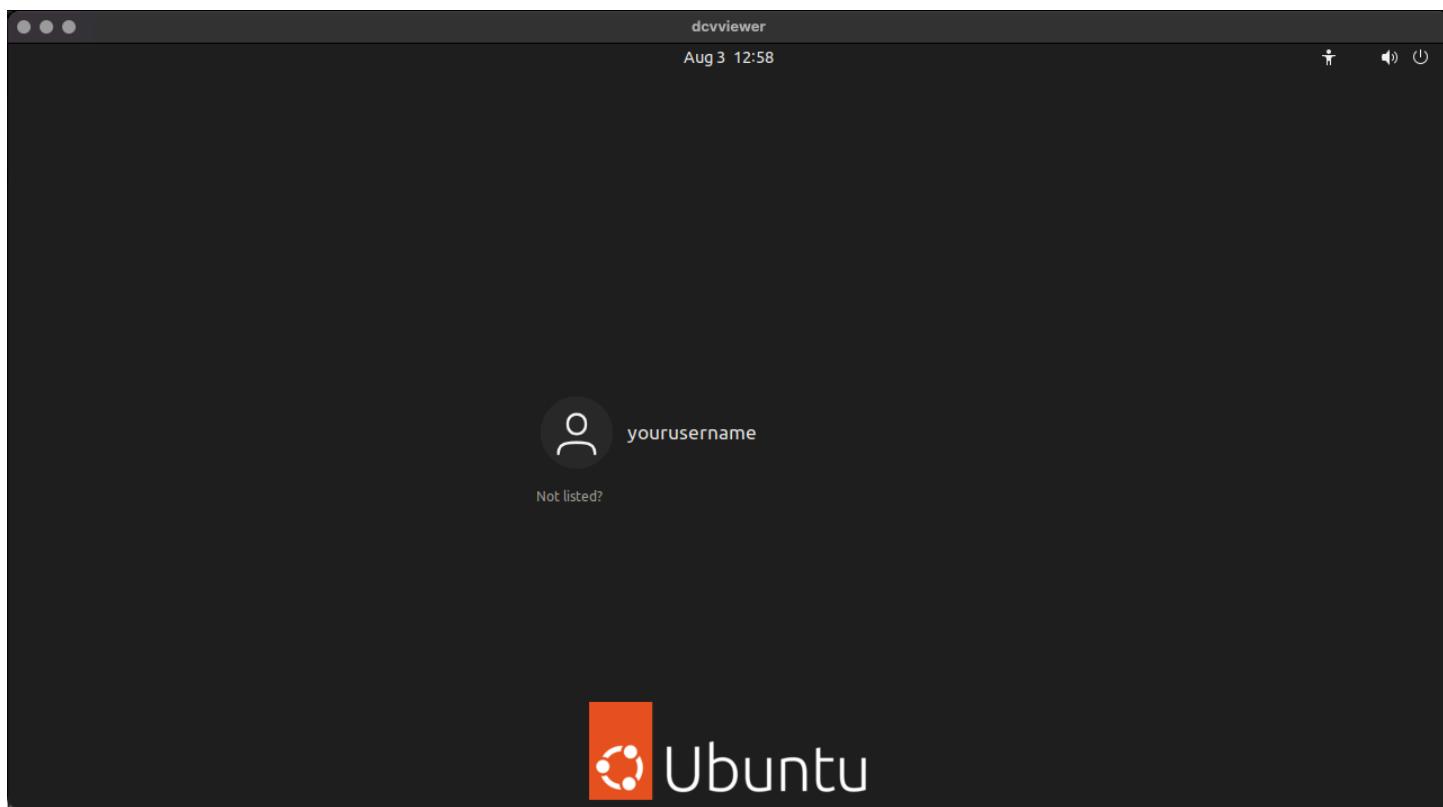
```
sudo systemctl start dcvserver
sudo systemctl enable dcvserver
```

Mit dem Amazon DCV-Client eine Verbindung zum Ubuntu-Server herstellen

Stellen Sie erneut eine Verbindung zu Ihrer Ubuntu-Instance her und erstellen Sie eine Sitzung für einen Benutzer, indem Sie Folgendes ausführen:

```
sudo dcv create-session --owner user --user user my-session --type console
```

Sie können jetzt den Amazon DCV-Client verwenden, um über deren öffentliche IP-Adresse auf Ihre Ubuntu-Instance zuzugreifen. Wenn Sie einen Amazon DCV-Client starten, wird ein Fenster angezeigt, in dem Sie über eine visuelle Anzeige auf Ihre Ubuntu-Instance zugreifen können.



Überprüfen Sie die GPU-Treiber

Stellen Sie sicher, dass die GPU-Treiber installiert sind und ordnungsgemäß funktionieren. Eine Möglichkeit, dies zu überprüfen, besteht darin, die [vkcube-Anwendung](#) in einem Terminal auszuführen.

1. Installieren Sie das `vulkan-tools` apt-Paket mit dem folgenden Befehl.

```
sudo apt install -y vulkan-tools
```

2. Führen Sie `vkcube`.

3. Überprüfen Sie die Ausgabe.

- Wenn Ihr System die richtige GPU ordnungsgemäß verwendet, erhalten Sie eine Ausgabe, die der folgenden ähnelt, mit dem Namen Ihrer GPU:
Selected GPU 0: AMD Radeon Pro V520 (RADV NAVI12), type: 2

- Wenn Ihre Anwendung die GPU nicht korrekt verwenden kann, wird möglicherweise eine andere Ausgabe angezeigt, die der folgenden ähnelt: Selected GPU 0: llvmpipe (LLVM 15.0.7, 256 bits), type: 4

Überprüfen Sie in diesem Fall die GPU-Treiber und installieren Sie sie bei Bedarf erneut.

Podman einrichten (nur Proton)

Wenn Sie eine Proton-Runtime verwenden, müssen Sie [Podman](#) installieren, einen Container, der vom Build-Prozess von Proton verwendet wird. Führen Sie die folgenden Schritte in einem Terminal aus.

1. Installieren Sie Podman, einen Container, den der Build-Prozess von Proton verwendet.

```
sudo apt install podman
```

2. In den Dateien und /etc/subgid /etc/subgid

- a. Stellen Sie sicher, dass in den Dateien der Benutzername und die ID Ihres Linux-Computers aufgeführt sind. Sie können entweder die Dateien öffnen oder den cat Befehl verwenden, um zu sehen, was in den Dateien enthalten ist. Beispiel für ein Format: test:165536:65536, wo Ihrem Benutzernamen test entspricht.
- b. Wenn sie nicht aufgeführt sind, fügen Sie sie hinzu. Beispiel für ein Format: test:165536:65536, wobei dies Ihrem Benutzernamen test entspricht.

```
$ cat /etc/subuid
      ceadmin:100000:65536
      test:165536:65536

$ cat /etc/subgid
      ceadmin:100000:65536
      test:165536:65536
```

Weitere Informationen finden Sie in der Podman-Dokumentation unter [Grundlegende Einrichtung und Verwendung von Podman in einer Rootless-Umgebung](#).

Nächster Schritt

Sie haben jetzt eine EC2 Amazon-Instance und -Umgebung eingerichtet, um Kompatibilitätsprobleme mit Amazon GameLift Streams zu beheben. Der nächste Schritt ist die Einrichtung von Proton. Anweisungen finden Sie unter [Fehlerbehebung bei Proton](#)

Fehlerbehebung bei der Kompatibilität mit Proton

In diesem Schritt richten Sie Proton auf Ihrem eigenen Computer ein, damit Sie Kompatibilitätsprobleme zwischen Ihrer Amazon GameLift Streams-Anwendung und Proton beheben können. Wenn Sie Ihre Anwendung in einer simulierten Umgebung ohne den Amazon GameLift Streams-Server ausführen, können Sie Probleme identifizieren, die für Ihre Anwendung und Laufzeitumgebung spezifisch sind.

Voraussetzungen

- Ubuntu 22.04 LTS mit installierten GPU-Treibern. Anweisungen finden Sie unter oder. [Richten Sie einen lokalen Computer ein](#) [Richten Sie einen Remote-Computer ein](#)

Installieren Sie Proton

[Um Proton auf Ihrem Ubuntu 22.04 LTS-Computer zu installieren, verwenden Sie das folgende Skript, um die Version von Proton, die Sie testen möchten, aus dem Proton-Repository zu klonen, zu erstellen und zu konfigurieren. GitHub](#)

1. Kopieren Sie den folgenden Code und fügen Sie ihn in eine Datei mit dem Namen `proton-setup.sh` auf Ihrem Ubuntu 22.04 LTS-Computer ein.

```
#!/bin/bash
# This is a script to build Proton. The default build is a tag from the
# experimental_9.0 branch of Proton, but can be changed as a parameter to this
# script.
#
# Usage: ./proton-setup.sh [optional proton_branch_name {default:
# experimental-9.0-20241121b}]
set -e

sudo apt install -y podman make git

# clone proton from github, recurse submodules
```

```
# if no proton git link is supplied, use a default tag from the experimental_8.0
branch
PROTON_BRANCH=${1:-"experimental-9.0-20241121b"}
PROTON_BUILD_DIR=protonBuild
PROTON_DIR=$(pwd)/proton
if git clone https://github.com/ValveSoftware/Proton.git --recurse-submodules --
branch $PROTON_BRANCH proton;
then
    echo "Successfully cloned Proton and its submodules."
else
    echo "Warning: a proton directory/repository already exists. It is recommended to
delete this folder and re-run this script unless it is a valid repository with
initialized submodules."
fi

if [ -d $PROTON_BUILD_DIR ];
then
    echo "Error: protonBuild directory already exists. Delete this folder first to
create a fresh build of Proton before re-running this script."
    exit 1
fi
mkdir $PROTON_BUILD_DIR
cd $PROTON_BUILD_DIR
$PROTON_DIR/configure.sh --enable-ccache --container-engine=podman

# build proton
echo "Building Proton"
make
echo "Done building Proton!"

# prepare proton for execution
cd dist
mkdir compatdata
if [ -e ./dist ]; then
    PROTON_FILES=dist
elif [ -e ./files ]; then
    PROTON_FILES=files
fi
cp version $PROTON_FILES/
echo "Finished installing proton. Proton binary location: $(pwd)/proton"
echo "STEAM_COMPAT_DATA_PATH: $(pwd)/compatdata"
echo "STEAM_COMPAT_CLIENT_INSTALL_PATH: anything"
```

2. In diesem Schritt führen Sie das Proton-Setup-Skript aus, um Proton und weitere Abhängigkeiten zu klonen und zu installieren. Das Skript akzeptiert als Argument den Tag- oder Branchnamen für die Proton-Version, die Sie installieren möchten. Verwenden Sie die nachstehenden Anweisungen für diese Version, um eine der von Amazon GameLift Streams bereitgestellten benutzerdefinierten Builds von Proton zu simulieren.

 Note

Rechnen Sie damit, dass das Klonen von einige Zeit GitHub in Anspruch nehmen wird. Es gibt viele Submodule zum Herunterladen, insgesamt mehrere Gigabyte.

Führen Sie in Ihrem Terminal das `proton-setup.sh` Skript aus und geben Sie den Proton-Versionszweig an:

- Integrierte Proton-Versionen
 - [Verwenden Sie für Proton 9.0-2 \(PROTON-20250516\) experimental-9.0-20241121b.](#)

`proton-setup.sh experimental-9.0-20241121b`

- [Verwenden Sie für Proton 8.0-5 \(PROTON-20241007\) experimental-8.0-20240205.](#)

`proton-setup.sh experimental-8.0-20240205`

In der Regel wird kein zusätzlicher Quellcode benötigt. [Wenn Sie jedoch Probleme mit Electra Media Player \(einem Unreal Engine-Plugin\) haben, empfehlen wir, die in https://github.com/ValveSoftware/wine/pull/257 enthaltenen Korrekturen zu verwenden.](#)

 Note

Für Proton 8.0-2c (PROTON-20230704) verwendet Amazon GameLift Streams einen proprietären Build, der nicht lokal erstellt werden kann.

- Empfohlene benutzerdefinierte Proton-Version

Für eine benutzerdefinierte Proton-Version empfehlen wir die Verwendung des Proton `experimental_8.0`-Zweigs.

```
proton-setup.sh experimental_8.0
```

- Andere benutzerdefinierte Proton-Versionen

Verwenden Sie für andere Proton-Versionen einen exakten Zweig- oder Tagnamen, der in [Proton-Versionen](#) aufgeführt ist.

```
proton-setup.sh branch-or-tag-name
```

Wenn Ihre Installation erfolgreich ist, sollte die Ausgabe in Ihrem Terminal der folgenden ähneln:

```
...
Done building Proton!
Finished preparing proton. Proton binary location: /home/test/protonBuild/dist/
proton
STEAM_COMPAT_DATA_PATH: /home/test/protonBuild/dist/compatdata
STEAM_COMPAT_CLIENT_INSTALL_PATH: anything
```

Notieren Sie sich die folgenden Variablen aus der Ausgabe, da Sie sie benötigen, um Proton im nächsten Schritt auszuführen:

- Binärlokation von Proton
- STEAM_COMPAT_DATA_PATH
- STEAM_COMPAT_CLIENT_INSTALL_PATH

Führen Sie Ihre Anwendung auf Proton aus

Bei den folgenden Schritten wird davon ausgegangen, dass sich die ausführbare Anwendung in path/myapplication/bin/application.exe befindet. Ersetzen Sie es durch den Pfad und den Dateinamen für Ihre Anwendung.

- Navigieren Sie in einem Terminal zu dem Ordner, in dem sich die ausführbare Datei Ihrer Anwendung befindet.

```
cd path/myapplication/bin/application.exe
```

- Führen Sie Ihre Anwendung auf Proton aus. Verwenden Sie den Proton-Binärspeicherort und die Umgebungsvariablen, die Sie im vorherigen Schritt erhalten haben.

```
STEAM_COMPAT_DATA_PATH=/home/test/protonBuild/dist/compatdata  
STEAM_COMPAT_CLIENT_INSTALL_PATH=anything /home/test/protonBuild/dist/proton run  
application.exe
```

Die Anwendung sollte jetzt versuchen zu starten. Wenn die Anwendung lokal, aber nicht in Amazon GameLift Streams gestartet wird, kann dies an einem Konfigurationsproblem beim Aufrufen von Amazon GameLift Streams liegen APIs. Stellen Sie sicher, dass die API-Aufrufparameter korrekt sind. Fahren Sie andernfalls mit dem nächsten Schritt zum Debuggen fort.

Debuggen Sie die Anwendung mithilfe von Protokolldateien

Wenn Ihre Anwendung Probleme beim Ausführen in der lokalen Proton-Umgebung hat, überprüfen Sie das Ausgabeprotokoll. Das Protokoll enthält Ausgaben aus Ihrer Anwendung und Laufzeitumgebung. Verfolgen Sie, wo Ihre Anwendung Probleme auf der Anwendungsseite nicht erkennt.

Verwenden Sie den folgenden Befehl, um die Protokollausgabe in eine Textdatei zu `proton.log` speichern, z. B.:

```
STEAM_COMPAT_DATA_PATH=/home/test/protonBuild/dist/compatdata  
STEAM_COMPAT_CLIENT_INSTALL_PATH=anything /home/test/protonBuild/dist/proton run  
application.exe &>proton.log
```

Proton gibt auch an, ob das Problem auf ein Wine-Plugin, eine nicht implementierte Funktion, fehlende DLLs usw. zurückzuführen ist. Weitere Informationen finden Sie im Wine-Debugging-Handbuch [von Wine HQ](#). Wenn Sie in den Protokollen einen Proton- oder Wine-Fehler finden, den Sie auf der Anwendungsseite nicht beheben können, wenden Sie sich an Ihren AWS Account Manager oder stellen Sie eine Frage in [AWS re:POST](#), um Hilfe beim weiteren Debuggen zu erhalten.

Profilierung der Leistung von Unreal Engine

In diesem Abschnitt erfahren Sie, wie Sie die Leistung Ihres Unreal Engine-Spiels oder Ihrer Anwendung analysieren können. Auf diese Weise können Sie Bereiche identifizieren, die optimiert werden müssen, was zu einem reibungsloseren Streaming in Amazon GameLift Streams führt.

Sie können die Konsole von Unreal Engine und die integrierten Statistikbefehle verwenden, um sich einen detaillierten Überblick über die Leistung Ihres Spiels zu verschaffen. Du kannst auf die Konsole in einem nicht lieferbaren Build oder im Editor zugreifen. Ein nicht lieferbarer Build bezieht sich auf ein Projekt, das mithilfe einer Debug- oder Entwicklungskonfiguration erstellt wurde.

Um auf die Konsole zuzugreifen

Drücken Sie in Builds, die nicht ausgeliefert werden können, und im Modus „[Im Editor abspielen](#)“ die Tilde-Taste (~), um die Konsole zu öffnen. Drücken Sie zweimal die Tilde-Taste, um die Konsole zu erweitern.

Hier sind einige Tipps zur Verwendung der Konsole:

- Geben Sie ein Schlüsselwort ein, um alle möglichen Befehle aufzulisten, die dieses Schlüsselwort enthalten. Scrollen Sie mit den Pfeiltasten durch die Liste.
- Scrollen Sie mit den Pfeiltasten oder den Tasten Bild auf und Bild ab durch den Verlauf.
- Protokolle werden in einer .txt Datei im Saved/Logs Verzeichnis Ihres Projekts gespeichert

Um ein Profil der Leistung deines Spiels zu erstellen

1. Führen Sie zunächst die stat unit Befehle stat fps und aus. Dadurch erhältst du einen Überblick darüber, wo dein Spiel mit der Leistung zu kämpfen hat.
 - stat fps: Zeigt die aktuellen Bilder pro Sekunde an.
 - stat unit: Unterteilt den Frame in mehrere Unterabschnitte.
 - Rahmen: Gesamtzeit der Wanduhr ab dem Zeitpunkt, zu dem die Simulation des Frames beginnt, bis zu dem Zeitpunkt, zu dem die Präsentation des Frames auf dem Bildschirm angezeigt wird.
 - Spiel: Gesamt-CPU-Zeit, die der Spielsimulations-Thread pro Frame beansprucht.
 - Zeichnen: Gesamt-CPU-Zeit für die Render-Threads, um die Szene in Befehle für die GPU zu übersetzen und diese an die GPU zu senden.
 - GPU: Gesamtzeit, die die GPU benötigt, um alle Befehle zu verarbeiten.
 - Ziehungen: Gesamtzahl der für den Frame eingereichten Ziehungen.
 - Prims: Gesamtzahl der gezeichneten Dreiecke.
2. Spiele das Spiel durch und identifiziere Bereiche mit geringer Leistung, was durch verringerte FPS und längere Zeit in Game, Draw oder GPU gekennzeichnet ist.

3. Laufstat game, um zu sehen, wie viel Zeit für die verschiedenen Spielgruppen aufgewendet wurde.
4. Verfeinere die Statistiken für bestimmte Gameplay-Faktoren wie KI, Animation, Physik, Gameplay, Scripting usw. Nachfolgend sind einige Beispiele aufgeführt:
 - stat ai: Zeit, das Verhalten der KI zu berechnen.
 - stat anim: Zeit für die Berechnung von Meshes mit Haut.
 - stat physics: Zeit, Physiksimulationen zu berechnen.
5. Führen Sie stat drawcount den Vorgang aus, um zu sehen, in welchen Renderbereichen die meisten Zeichnungen generiert werden. Die Liste zeigt die Renderdurchgänge, die Draws ausgeben, und die Anzahl der Draws, die pro Frame ausgegeben wurden. Weitere Informationen erhalten Sie, indem Sie im nächsten Schritt die GPU-Statistiken analysieren.
6. Führen Sie das Programm aus, stat gpu um zu sehen, welche Rendertypen die meiste GPU-Zeit verbrauchen.
7. Unterteilen Sie die Rendertypen in große Gruppen wie Lichter, Schatten, Lumen (Beleuchtung), Haare, Nachbearbeitung usw. Hier sind ein paar gängige Beispiele:
 - stat lightrendering: GPU-Zeit zum Rendern von Lichtern und Schatten.
 - stat shadowrendering: GPU-Zeit zum Aktualisieren der verschiedenen Schatten.
 - stat scenerendering: GPU-Zeit zum Rendern der Szene.

Dieser Abschnitt behandelt nur eine Teilmenge der verfügbaren Befehle. Abhängig von den Funktionen Ihres Spiels sollten Sie sich Statistiken für Bereiche wie Asset-Streaming, virtuelle Texturierung, Verteilung der CPU-Arbeitslast, Threading, Sound, Partikel usw. ansehen. Weitere Informationen finden Sie unter [Stat-Befehle](#).

Regionen, Kontingente und Beschränkungen

Amazon GameLift Streams ist auf mehreren AWS-Regionen Plattformen verfügbar und bietet Dual-Stack-Serviceendpunkte, die IPv4 sowohl IPv6 Konnektivität als auch Konnektivität unterstützen. Der Service wird von Hauptstandorten wie USA Ost (Ohio), USA West (Oregon), Asien-Pazifik (Tokio) und Europa (Frankfurt) aus betrieben AWS-Regionen und bietet die Möglichkeit, zusätzliche Standorte zu verwalten, die zusammen als Remote-Standorte bezeichnet werden, um Latenz und Streamqualität zu optimieren.

Die Serviceinfrastruktur unterliegt drei Hauptkategorien von Einschränkungen:

- Servicekontingente
- API-Ratenlimits
- Dienstbeschränkungen wurden behoben

Dazu gehören Einschränkungen in Bezug auf Anwendungsgrößen, Anzahl der Anwendungen pro Region, Dateiverwaltungskapazitäten und GPU-Zuweisungen für verschiedene Stream-Klassen und Regionen. Der Dienst implementiert spezifische API-Ratenbegrenzungen für verschiedene Operationen, die zwischen 1 und 20 Anfragen pro Sekunde liegen, um eine stabile Serviceleistung zu gewährleisten. Darüber hinaus gibt es feste Dienstbeschränkungen in Bezug auf Stream-Gruppenkonfigurationen, GPU-Bereitstellungen und Anwendungszuordnungen, die für alle Kunden einheitlich gelten.

AWS-Regionen und entfernte Standorte, die von Amazon GameLift Streams unterstützt werden

An AWS-Region ist eine Sammlung von AWS Ressourcen in einem geografischen Gebiet. Jede AWS-Region ist isoliert und unabhängig von den anderen Regionen. Allgemeine Informationen zu AWS-Regionen finden Sie unter [Verwaltung AWS-Regionen](#) in der Allgemeine AWS-Referenz.

In der folgenden Tabelle sind aufgeführt AWS-Regionen , wo der Amazon GameLift Streams-Service verfügbar ist, und die Endpunkte für jede Region. Sie erstellen alle Amazon GameLift Streams-Anwendungs- und Stream-Gruppenressourcen in einer bestimmten Region, unabhängig davon, ob Sie in der Amazon GameLift Streams-Konsole arbeiten, AWS Command Line Interface (AWS CLI) verwenden oder programmatische Aufrufe tätigen. Die Region, in der Sie diese Ressourcen erstellen,

wird als primärer Standort bezeichnet. Verwenden Sie den Endpunkt Ihres primären Standorts, um programmgesteuert eine Verbindung zum Amazon GameLift Streams-Service herzustellen.

Service-Endpunkte

Amazon GameLift Streams unterstützt Dual-Stack-Serviceendpunkte, sodass Kunden und Ressourcen mithilfe IPv6 von oder mit dem Service interagieren können. IPv4

Name der Region	Region	Endpunkt	Protocol (Protokoll)
USA Ost (Ohio)	us-east-2	gameliftstreams.us-east-2.api.aws	HTTPS
USA West (Oregon)	us-west-2	gameliftstreams.us-west-2.api.aws	HTTPS
Asien-Pazifik (Tokio)	ap-northeast-1	gameliftstreams.ap-northeast-1.api.aws	HTTPS
Europa (Frankfurt)	eu-central-1	gameliftstreams.eu-central-1.api.aws	HTTPS

Entfernte Standorte

Amazon GameLift Streams kann die Abdeckung auf entfernte Standorte ausdehnen, sodass Sie Ihre Anwendung und Stream-Sitzungen an mehr Orten hosten können. Welche Remote-Standorte Ihnen zur Verfügung stehen, hängt von Ihrem primären Standort ab. Wir empfehlen Ihnen, Standorte auszuwählen, die sich geografisch in der Nähe Ihrer Benutzer befinden, um die Latenz und die Streamqualität zu optimieren.

Primärer Standort	Abgelegene Standorte
US East (Ohio) – us-east-2	<ul style="list-style-type: none"> • USA Ost (Nord-Virginia) – us-east-1 • USA West (Oregon) – us-west-2 • Europa (Ireland) – eu-west-1 • Europa (Frankfurt) – eu-central-1 • Asien-Pazifik (Tokio) – ap-northeast-1
USA West (Oregon) – us-west-2	<ul style="list-style-type: none"> • USA Ost (Nord-Virginia) – us-east-1 • USA Ost (Ohio) – us-east-2 • Europa (Ireland) – eu-west-1 • Europa (Frankfurt) – eu-central-1 • Asien-Pazifik (Tokio) – ap-northeast-1
Asien-Pazifik (Tokio) – ap-northeast-1	<ul style="list-style-type: none"> • USA Ost (Nord-Virginia) – us-east-1 • USA West (Oregon) – us-west-2 • USA Ost (Ohio) – us-east-2

Primärer Standort	Abgelegene Standorte
	<ul style="list-style-type: none"> • Europa (Ireland) – eu-west-1 • Europa (Frankfurt) – eu-central-1
Europa (Frankfurt) – eu-central-1	<ul style="list-style-type: none"> • USA Ost (Nord-Virginia) – us-east-1 • USA West (Oregon) – us-west-2 • USA Ost (Ohio) – us-east-2 • Europa (Ireland) – eu-west-1 • Asien-Pazifik (Tokio) – ap-northeast-1

Unterstützte Standorte nach Stream-Klassen in Amazon GameLift Streams

Die folgende Tabelle zeigt die Verfügbarkeit der einzelnen Stream-Klassenfamilien an allen unterstützten AWS-Regionen und entfernten Standorten.

Name der Region	Region	Gen6*	Gen5*	Gen4*
Asien-Pazifik (Tokio)	ap-northeast-1	✓ Ja	✓ Ja	✓ Ja
Europa (Frankfurt)	eu-central-1	✓ Ja	✓ Ja	✓ Ja
Europa (Irland)	eu-west-1	✗ Nein	✓ Ja	✓ Ja
USA Ost (Nord-Virginia)	us-east-1	✓ Ja	✓ Ja	✓ Ja

Name der Region	Region	Gen6*	Gen5*	Gen4*
USA Ost (Ohio)	us-east-2	✓ Ja	✓ Ja	✓ Ja
USA West (Oregon)	us-west-2	✓ Ja	✓ Ja	✓ Ja

Amazon GameLift Streams-Servicekontingente

Service Quotas, auch als Limits bezeichnet, sind die maximale Anzahl von Serviceressourcen oder -vorgängen für Ihr AWS-Konto.

Viele der Service-Kontingente in Amazon GameLift Streams begrenzen die Gesamtzahl der GPUs (Rechenressourcen), die Sie für das Streaming in Ihrem Konto konfigurieren können. Genauer gesagt geben diese GPU-Servicekontingente die maximale GPUs Anzahl einer bestimmten Stream-Klassenfamilie an, die Sie pro Standort für alle Stream-Gruppen in Ihrem Konto anfordern können. Wenn Ihr Konto beispielsweise ein Limit von 5 gen5n GPUs Zoll hatus-west-2, muss die Summe der Daten, die zur Bereitstellung der gesamten Stream-Kapazität us-west-2 für alle Ihre Stream-Gruppen gen5n GPUs erforderlich sind, kleiner oder gleich 5 sein. Dies gilt sowohl GPUs für ständig verfügbare als auch für On-Demand-Kapazität.

Weitere Informationen darüber, wie Kontingente mit der Stream-Kapazität interagieren, finden Sie unter. [Kapazität und Servicekontingenten](#) Informieren Sie [API-Ratenlimits](#) sich auch über zusätzliche Einschränkungen, die Sie in Amazon GameLift Streams beachten sollten. [Andere Einschränkungen](#)

Zeigen Sie Ihr standardmäßiges oder angewendetes Kontingent auf Kontoebene und die Auslastung in der Konsole Service Quotas an, indem Sie GameLift Streams als AWS Service auswählen.

Allgemeine Informationen zu Servicekontingenten finden Sie unter [AWS Servicekontingenten](#) im Allgemeine AWS-Referenz.

Servicekontingente

In der folgenden Tabelle sind die GPU-Kontingente standardmäßig alle auf 0 gesetzt. Die für Ihr Konto angewendeten Kontingente können jedoch unterschiedlich sein. Um dies zu überprüfen, melden Sie sich [bei Amazon GameLift Streams](#) an AWS-Managementkonsole und öffnen Sie die Service-Kontingents-Konsole. Dort können Sie Ihre aktuellen Kontingente in der Spalte Angewandter Kontingentwert auf Kontoebene und die Nutzung dieser Kontingente in der Spalte Auslastung überprüfen und eine Anfrage zur Erhöhung dieser Werte einreichen.

Name	Standard	Anpassbar	Description
Anwendungsgröße (GiB)	Jede unterstützte Region: 100	<u>Yes</u> (Ja)	Die maximale Gesamtgröße (in GiB) einer Anwendung in diesem Konto. Beachten Sie, dass ein Gibibyte (GiB) $1024 * 1024 * 1024$ Byte entspricht.
Anwendungen	Jede unterstützte Region: 20	<u>Ja</u>	Die maximale Anzahl von Anwendungen, die Sie in diesem Konto erstellen können, pro Region. AWS
Dateien pro Anwendung	Jede unterstützte Region: 30.000	<u>Ja</u>	Die maximale Anzahl von Dateien, die Sie in einer Anwendung in diesem Konto haben können.
Gen4n GPUs, ap-northeast-1	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN4n, die GPUs Sie für das Streaming am Standort ap-northeast-1 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN4N_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.
Gen4n GPUs, eu-central-1	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN4n, die GPUs Sie für das Streaming am

Name	Standard	Anpas	Description
			Standort eu-central-1 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN4n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.
Gen4n GPUs, eu-west-1	Jede unterstützte Region: 0	Ja	Die maximale Anzahl von GEN4n, die GPUs Sie für das Streaming am Standort eu-west-1 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN4n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.

Name	Standard	Anpas	Description
Gen4n GPUs, us-east-1	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN4n, die GPUs Sie für das Streaming am Standort us-east-1 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN4n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.
Gen4n GPUs, us-east-2	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN4n, die GPUs Sie für das Streaming am Standort us-east-2 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN4n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.

Name	Standard	Anpas	Description
Gen4n GPUs, US-West-2	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN4n, die GPUs Sie für das Streaming am Standort US-West-2 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN4n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.
Gen5n GPUs, ap-northeast-1	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN5n, die GPUs Sie für das Streaming am Standort ap-northeast-1 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN5n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.

Name	Standard	Anpas	Description
Gen5n GPUs, eu-central-1	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN5n, die GPUs Sie für das Streaming am Standort eu-central-1 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN5n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.
Gen5n GPUs, eu-west-1	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN5n, die GPUs Sie für das Streaming am Standort eu-west-1 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN5n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.

Name	Standard	Anpas	Description
Gen5n GPUs, us-east-1	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN5n, die GPUs Sie für das Streaming am Standort us-east-1 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN5n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.
Gen5n GPUs, us-east-2	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN5n, die GPUs Sie für das Streaming am Standort us-east-2 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN5n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.

Name	Standard	Anpas	Description
Gen5n GPUs, US-West-2	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN5n, die GPUs Sie für das Streaming am Standort US-West-2 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN5n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.
Gen6n GPUs, ap-northeast-1	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN6n, die GPUs Sie für das Streaming am Standort ap-northeast-1 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN6n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.

Name	Standard	Anpas	Description
Gen6n GPUs, eu-central-1	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN6n, die GPUs Sie für das Streaming am Standort eu-central-1 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN6n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.
Gen 6n GPUs, us-east-1	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN6n, die GPUs Sie für das Streaming am Standort us-east-1 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN6n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.

Name	Standard	Anpas	Description
Gen 6n GPUs, us-east-2	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN6n, die GPUs Sie für das Streaming am Standort us-east-2 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN6n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.
Gen 6n GPUs, US-West-2	Jede unterstützte Region: 0	<u>Ja</u>	Die maximale Anzahl von GEN6n, die GPUs Sie für das Streaming am Standort US-West-2 für alle Stream-Gruppen in diesem Konto konfigurieren können. Stream-Klassen mit mehreren Mandanten, wie z. B. „GEN6n_High“, unterstützen das Streamen von mehr als einer Sitzung pro GPU.

Name	Standard	Anpas	Description
Stream-Gruppen	Jede unterstützte Region: 5	<u>Yes</u> (Ja)	Die maximale Anzahl von Stream-Gruppen, die Sie in diesem Konto pro Region erstellen können. AWS Eine Stream-Gruppe ist eine Sammlung von Rechenressourcen, die Ihre Anwendung an Endbenutzer streamen.

Amazon GameLift Streams API-Ratenbegrenzungen

Diese Grenzwerte spiegeln die maximale Rate von Anfragen pro Sekunde von Ihrem AWS-Konto an den Amazon GameLift Streams-Service in einem AWS-Region.

API-Operation	Anforderungen pro Sekunde
AddStreamGroupLocations	5
AssociateApplications	5
CreateApplication	5
CreateStreamGroup	1
CreateStreamSessionConnection	20
DeleteApplication	5
DeleteStreamGroup	5
DisassociateApplications	5
ExportStreamSessionFiles	20
GetApplication	10

API-Operation	Anforderungen pro Sekunde
GetStreamGroup	10
GetStreamSession	20
ListApplications	10
ListStreamGroups	10
ListStreamSessions	20
ListStreamSessionsByAccount	20
ListTagsForResource	10
RemoveStreamGroupLocations	5
StartStreamSession	20
TagResource	10
TerminateStreamSession	20
UntagResource	10
UpdateApplication	5
UpdateStreamGroup	5

Andere Einschränkungen von Amazon GameLift Streams

Auf dieser Seite sind weitere Einschränkungen aufgeführt, die Sie bei der Erstellung Ihrer Streaming-Lösung beachten sollten. Diese Beschränkungen sind im Service für alle Kunden festgelegt.

Name	Einschränkung	Description
Anwendungen in einer Stream-Gruppe	100	Die maximale Anzahl von Amazon GameLift

Name	Einschränkung	Description
		Streams-Anwendungen, die einer Stream-Gruppe zugeordnet werden können.
GPUs in einer Stream-Gruppe	2500	Die maximale Anzahl von Mitgliedern einer Stream-Gruppe GPUs in allen Regionen und abgelegenen Standorten.
Größe einer einzelnen Datei (GiB)	80 GiB	Die maximale Größe (in GiB) einer einzelnen Datei in einer Anwendung. Beachten Sie, dass ein Gibibyte (GiB) $1024^3 \times 1024^3$ * 1024 Byte entspricht.
Stream-Gruppenzuordnungen pro Anwendung	100	Die maximale Anzahl von Stream-Gruppen, denen eine Amazon GameLift Streams-Anwendung zugeordnet werden kann.

Nutzung und Rechnungen für Amazon GameLift Streams verwalten

In diesem Thema erfahren Sie, wie Sie Ihre Nutzung, Kosten und Abrechnung von Amazon GameLift Streams überwachen und verwalten können, um Ihre Streaming-Ausgaben zu optimieren.

Die folgenden Informationen finden Sie auch auf der [Seite mit den Amazon GameLift Streams-Preisen](#):

- Kostenaufschlüsselung: Erfahren Sie, welche AWS Gebühren Ihnen bei der Nutzung von Amazon GameLift Streams entstehen.
- Amazon GameLift Streams-Tarife: Erfahren Sie, wie viel Amazon GameLift Streams kostet, und vergleichen Sie verschiedene Optionen.
- Reservierung von Stream-Kapazitäten: Planen Sie im Voraus und stellen Sie sicher, dass Sie über genügend Stream-Kapazität verfügen, um die Anforderungen Ihrer Kunden zu erfüllen.

Überprüfen Sie Ihre Amazon GameLift Streams-Rechnungen und Ihre Nutzung

Sie können Ihre Amazon GameLift Streams-Rechnungen und die Nutzung überprüfen, indem Sie die AWS Fakturierung und Kostenmanagement Tools in der AWS Konsole oder verwenden AWS CLI.

Informationen zur Anzeige Ihrer Rechnung in der AWS Konsole finden Sie unter [Ihre Rechnung anzeigen](#) im AWS Billing Benutzerhandbuch.

Um Ihre Rechnung über einzusehen AWS CLI, rufen Sie [GetCostAndUsage](#) über die Billing and Cost Management API an. Verwenden Sie beispielsweise den folgenden Befehl, um eine monatliche Rechnung für Amazon GameLift Streams abzurufen, und ersetzen Sie die Daten durch die für Sie relevanten Daten.

Example : Verwenden Sie die **GetCostAndUsage** API, um die Rechnung anzuzeigen

```
aws ce get-cost-and-usage /  
  --time-period Start=2023-01-01,End=2023-01-31 /  
  --granularity MONTHLY /  
  --metrics BlendedCost /  
  --filter Amazon GameLift Streams-bill-filter.json
```

wobei der Filter, z. B. `Amazon GameLift Streams-bill-filter.json`, den Amazon GameLift Streams-Service wie folgt spezifiziert:

```
{  
    "Dimensions": {  
        "Key": "SERVICE",  
        "Values": ["Amazon GameLift Streams"]  
    }  
}
```

Bewährte Methoden zur Verwaltung der Amazon GameLift Streams-Kosten

Wir empfehlen Ihnen dringend, die folgenden Tools und Techniken zur Verwaltung Ihrer Amazon GameLift Streams-Kosten zu verwenden, um unerwartete Kosten zu vermeiden.

Erstellen Sie Abrechnungsbenachrichtigungen, um die Nutzung zu überwachen

Richten Sie mithilfe von AWS Budgets Abrechnungswarnungen ein, mit denen Sie Ihre Kosten und Nutzung verfolgen und schnell auf Benachrichtigungen reagieren können, um unerwartete Kosten zu vermeiden. Sie können die Abrechnungswarnung auch so konfigurieren, dass sie Aktionen auslöst, die Ihnen helfen, das Budget einzuhalten. Standardmäßig beinhalten Budgets alle Ihre AWS-Services. Um ein Budget nur für Amazon GameLift Streams anzugeben, fügen Sie einen [Budgetfilter](#) hinzu.

Weitere Informationen finden Sie unter den folgenden Themen:

- [Erstellen eines Budgets](#)
- [Bewährte Methoden für AWS Budgets](#)

Skalieren Sie Stream-Gruppen auf eine Kapazität von Null

Für zugewiesene Stream-Kapazität fallen weiterhin Kosten an, auch wenn sie derzeit keine Stream-Sitzungen hosten. Skalieren Sie Stream-Gruppen auf Kapazität Null, wenn sie nicht genutzt werden, um unnötige Kosten zu vermeiden. Dadurch wird verhindert, dass Ihre Stream-Gruppe Ressourcen zuweist. Wenn Sie die Stream-Kapazität für Always-On und On-Demand-Streams auf

Null setzen, enden alle verbundenen Streams. Wenn Sie bereit sind, können Sie Ihre Stream-Gruppe wiederverwenden, indem Sie die Kapazität wieder skalieren.

Eine Anleitung dazu finden Sie unter [Kapazität bearbeiten](#).

 Warning

Vermeiden Sie es, eine Stream-Gruppe zu löschen, es sei denn, Sie haben nicht vor, die Stream-Gruppe erneut zu verwenden. Wenn Sie eine Stream-Gruppe löschen, können Sie die ursprüngliche Stream-Gruppe nicht wiederherstellen und müssen eine neue erstellen.

Löschen Sie die ursprünglichen Anwendungsdateien

Um die Speicherkosten zu optimieren, können Sie die ursprünglichen Anwendungsdateien löschen, die Sie in einen Amazon S3 S3-Bucket hochgeladen haben. Es ist sicher, die Dateien zu löschen, wenn sich die Anwendung im Status Bereit befindet. Zu diesem Zeitpunkt hat Amazon GameLift Streams einen Snapshot der Anwendungsdateien und greift nicht mehr auf Ihre Originaldateien zu.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.