

デベロッパーガイド

Amazon GameLift Streams



Amazon GameLift Streams: デベロッパーガイド

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

Amazon の商標およびトレードドレスは Amazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していない他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

Amazon GameLift Streams とは	1
機能	1
Amazon GameLift Streams の使用を開始する方法	2
Amazon GameLift ストリームへのアクセス	2
利用規約	3
設定	4
にサインアップする AWS アカウント	4
管理アクセスを持つユーザーを作成する	5
プログラムによるアクセスの取得	6
Web SDK のダウンロード	7
のダウンロード AWS CLI	7
請求アラートを設定する	8
はじめに	9
設定の選択	9
開始点	10
コストの最適化	11
設定の決定	12
設定の選択が次のステップにどのように影響するか	13
次の手順	14
設定オプション	14
ランタイム環境	14
ストリームクラス	15
ランタイム環境とストリームクラスの互換性	21
最初のストリーム	22
前提条件	23
ステップ 1: アプリケーションを Amazon S3 バケットにアップロードする	23
ステップ 2: Amazon GameLift Streams 用にアプリケーションを設定する	25
ステップ 3: Amazon GameLift Streams がアプリケーションをストリーミングする方法を管理する	30
ステップ 4: Amazon GameLift Streams でストリームをテストする	37
ステップ 5: クリーンアップする (スキップしない)	38
ストリームの管理	40
主要なコンセプト	40
アプリケーション	42

アップロードする前に	42
Amazon S3 バケットにアプリケーションをアップロードする	44
アプリケーションの作成	45
アプリケーションを編集する	51
アプリケーションの削除	54
アプリケーションログバケットのアクセス許可ポリシー	56
リンクされたストリームグループ	57
ストリームグループ	57
ストリーム容量について	59
キャパシティとサービスクォータ	61
ロケーションについて	62
ストリームグループを作成する	63
全般設定の編集	75
容量の編集	76
容量のスケールダウン動作	77
ストリームグループにロケーションを追加する	78
ストリームグループ内の場所を削除する	79
ストリームグループを削除する	81
リンクされたアプリケーション	82
ストリームグループのライフサイクル	82
ストリームグループのメンテナンス	83
マルチアプリケーションストリームグループ	83
制限事項と要件	84
デフォルトアプリケーションについて	85
デフォルトのアプリケーションを変更する	85
アプリケーションをリンクする	87
アプリケーションのリンクを解除する	89
ストリームセッション	91
ストリームセッションについて	91
コンソールでのストリームのテスト	91
ストリームセッションのライフサイクル	93
ストリームセッションに影響するタイムアウト値	95
ストリームセッションの終了	96
ストリームセッションへの再接続	96
ストリームセッションファイルのエクスポート	97
仕組み	97

コストへの影響	98
ファイルのエクスポート (コンソール)	99
ファイルのエクスポート (CLI)	99
Amazon GameLift Streams バックエンドサービスとウェブクライアント	102
サポートされているブラウザと入力	102
既知の問題	104
制限事項	104
IPv6 サポート	105
必要なポート	105
ウェブサーバーとクライアントのセットアップ	106
前提条件	106
Web SDK のダウンロード	106
ストリーミングリソースをセットアップする	107
バックエンドサーバーをセットアップする	107
ウェブクライアントを起動する	108
ストリーミングリソースをクリーンアップする	109
ストリームの外観をカスタマイズする	110
画面のロード	110
ロケール設定	111
マウスの移動処理	112
マウス入力モード	112
ポインタロック	113
ベストプラクティス	114
データチャネル通信	114
特徴	115
データチャネルの使用	115
クライアント側	116
アプリケーション側	116
起動チェックリスト	120
Amazon GameLift Streams チームに通知する	120
互換性とパフォーマンスのテスト	120
キャパシティ予約	120
大規模なパフォーマンステスト	121
起動前セットアップ	121
その他のヒント	121
さらにサポートが必要ですか？	122

セキュリティ	123
データ保護	124
保管中の暗号化	126
転送中の暗号化	126
エンドユーザーストリームの保護	126
Linux ストリームクラスのセッション分離	127
Windows ストリームクラスのセッション分離	127
暗号化キーの管理	127
ネットワーク間トラフィックのプライバシー	128
Identity and Access Management	128
オーディエンス	128
アイデンティティを使用した認証	129
ポリシーを使用したアクセスの管理	130
Amazon GameLift Streams と IAM の連携方法	132
アイデンティティベースのポリシーの例	138
トラブルシューティング	141
コンプライアンス検証	142
耐障害性	142
インフラストラクチャセキュリティ	142
再利用とマルチテナント	143
インターフェイス VPC エンドポイント	144
設定と脆弱性の分析	146
セキュリティのベストプラクティス	147
Amazon GameLift ストリームのモニタリング	148
CloudWatch を使用して監視する	148
ストリームグループの容量と使用状況	149
ストリームグループのパフォーマンスとリソース使用率	150
ストリームステータス	151
カスタマーエンゲージメント	152
データチャネル	152
CloudTrail による API コールのログ記録	154
CloudTrail で Amazon GameLift Streams データイベント	155
CloudTrail での Amazon GameLift Streams 管理イベント	156
Amazon GameLift Streams イベントの例	157
リアルタイムのパフォーマンス統計	161
パフォーマンス統計を受信する	162

パフォーマンス統計リファレンス	164
トラブルシューティング	166
アクセスが拒否されました	166
アプリケーションに関する問題	167
事前チェック	167
Proton の問題	167
画面解像度によるアプリケーションの問題	168
ストリームセッションの開始時にアプリケーションが終了する	168
Unreal Engine アプリケーションがクラッシュするか、追加の依存関係が必要です	168
パフォーマンスの問題	169
Amazon GameLift Streams でストリーミングするとゲームのパフォーマンスが低下する	169
Windows アプリケーションでは、ロード時間が遅い、またはスタッタリングの問題が発生する	170
ストリーム接続の問題	176
ストリーム入力の問題	177
一般的な入力のトラブルシューティング	177
ゲームパッドとマイクの入力がネイティブ Linux アプリケーションで機能しない	177
MacOS クライアントでキー入力がスタックしているように見える	177
マウスの動きは Amazon GameLift Streams で動作が異なります	178
ストリームセッションの問題	178
ストリームセッションが開始されない	179
ストリームセッションが終了しました	180
Proton との互換性	180
テストとトラブルシューティングの大まかな手順	180
Proton の既知の問題	181
ローカルマシンをセットアップする	181
リモートマシンをセットアップする	182
Proton でのトラブルシューティング	187
Unreal Engine のパフォーマンスのプロファイリング	192
リージョン、クオータ、制限	195
AWS リージョン およびリモートロケーション	195
サービスエンドポイント	196
リモートロケーション	196
ストリームクラスでサポートされているロケーション	198
サービスクオータ	199
サービスクオータ	200

API レート制限	209
その他の制限事項	211
使用量と請求	213
Amazon GameLift Streams の請求書と使用状況を確認する	213
Amazon GameLift Streams のコストを管理するためのベストプラクティス	214
請求アラートを作成して、使用状況をモニタリングする	214
ストリームグループを容量ゼロにスケールする	214
元のアプリケーションファイルを削除する	215

CCXVI

Amazon GameLift Streams とは

Amazon GameLift Streams を使用すると、ゲームパブリッシャーなどは、世界中のプレイヤーや視聴者に低レイテンシーのオンデマンドストリーミングエクスペリエンスを提供できます。Amazon GameLift Streams は、独自のストリーミングテクノロジーと AWS グローバルインフラストラクチャを組み合わせて、大規模なアプリケーションストリーミングを運用および維持します。パブリッシャーは、容量とコストを効果的に管理するために、オンデマンドストリーミングリソースとリザーブドストリーミングリソースの両方を柔軟にプロビジョニングできます。

トピック

- [機能](#)
- [Amazon GameLift Streams の使用を開始する方法](#)
- [Amazon GameLift ストリームへのアクセス](#)

機能

Amazon GameLift Streams には、次の主要な機能があります。

- AWS グローバルフットプリントを使用して、ブラウザを使用する任意のデバイスにplayer-to-cloudレイテンシーを最小限に抑えながら、リアルタイムのゲームプレイ体験を提供するストリーミングテクノロジー。
- ダウンロードを必要とせずに、高解像度 (1080p) 解像度と 60 fps でシームレスにゲームをプレイし、ブラウザベースのデバイスを強力なゲームマシンに変換します。
- 顧客の需要に合わせてストリーミング容量を調整するスケーリングツール。例えば、これらのツールを使用すると、新しいプレイヤーをストリームセッションにすばやく対応させるのに十分な容量を維持しながら、ゲームストリーミングコストを調整できます。
- Amazon GameLift Streams コンソールを使用してパフォーマンス分析をストリーミングし、メトリクスの追跡、ストリームログの表示、ストリームリソースの使用状況に関するデータの確認を行います。
- Windows および Linux ベースのゲームをほとんどまたはまったく変更せずに直接ストリーミングします。
- Amazon GameLift Streams SDK は、既存の ID サービス、ストアフロント、クライアントアプリケーションの統合に役立ちます。

Amazon GameLift Streams の使用を開始する方法

Amazon GameLift Streams を初めて使用する場合は、以下のトピックから始めることをお勧めします。

- [開発者としての Amazon GameLift Streams のセットアップ](#) では、ユーザーアクセス権 AWS アカウントを持つ者の取得や Amazon GameLift Streams での開発に必要なソフトウェアの設定など、1回限りのセットアップタスクについて説明します。
- [Amazon GameLift Streams で最初のストリームを開始する](#) は、コンテンツストリーミングワークフローの重要なステップをガイドします。ゲームビルドなどのコンテンツから、Amazon GameLift Streams ストリーミングクラウドリソースをプロビジョニングし、ストリーミングセッションを開始します。

Amazon GameLift ストリームへのアクセス

以下のツールを使用して、アプリケーションコンテンツとストリーミングリソースを作成、アクセス、管理できます。

- AWS マネジメントコンソール – Amazon GameLift Streams アプリケーションとストリームグループの作成と管理に使用できるウェブインターフェイスを提供します。
- AWS Command Line Interface (AWS CLI) – Windows、Mac、Linux でサポートされている 幅広いセットのコマンド AWS のサービス を提供します。このツールの詳細については、[AWS Command Line Interface 「」ページ](#)を参照してください。
- AWS SDK – 言語固有の APIs を提供し、署名の計算、リクエストの再試行の処理、エラー処理など、接続の詳細を処理します。Amazon GameLift Streams サービス API のドキュメントについては、[「Amazon GameLift Streams API リファレンス」](#)を参照してください。AWS SDK の一般的な情報については、[「構築するツール AWS」](#)を参照してください。

サポートされている の詳細については AWS リージョン、「」を参照してください[リージョン、クォータ、制限。](#)

Amazon GameLift Streams の利用規約

Amazon GameLift Streams を使用する前に、ストリーミングするアプリケーションに適用されるライセンス条項やストリーミングする場所など、適用されるすべての法的要件に準拠できることを確認してください。

- AWS 要件の詳細については、[AWS サービス条件](#)のセクション 43 を参照してください。
- サービスレベルアグリーメントの詳細については、[「Amazon GameLift Streams サービスレベルアグリーメント」](#)を参照してください。

開発者としての Amazon GameLift Streams のセットアップ

プロジェクトで Amazon GameLift Streams サービスの使用を開始するには、以下の基本的なセットアップタスクを完了します。Amazon GameLift Streams で使用する AWS アカウント とそのアカウントの下にユーザーが既に存在する場合は、「」にスキップできます[Web SDK のダウンロード](#)。

でできることの詳細については AWS アカウント、[「AWS の開始方法」](#) を参照してください。

これらのセットアップタスクを完了したら、[Amazon GameLift Streams で最初のストリームを開始する](#) 「」に進み、チュートリアルを順を追って実行することをお勧めします。チュートリアルでは、ウェブクライアントでコンテンツストリーミングを取得するためのワークフロー全体について説明します。

トピック

- [にサインアップする AWS アカウント](#)
- [管理アクセスを持つユーザーを作成する](#)
- [プログラムによるアクセスの取得](#)
- [Amazon GameLift Streams Web SDK をダウンロードする](#)
- [のダウンロード AWS CLI](#)
- [請求アラートを設定する](#)

にサインアップする AWS アカウント

がない場合は AWS アカウント、次の手順を実行して作成します。

にサインアップするには AWS アカウント

1. <https://portal.aws.amazon.com/billing/signup> を開きます。
2. オンラインの手順に従います。

サインアップ手順の一環として、電話またはテキストメッセージを受け取り、電話キーパッドで検証コードを入力します。

にサインアップすると AWS アカウント、AWS アカウントのルートユーザー が作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権があります。セキュリティベストプラクティスとして、ユーザーに管理アクセス権を割り当て、[ルートユーザーの管理](#) を実行します。

トユーザーアクセスが必要なタスクの実行にはルートユーザーのみを使用するようにしてください。

AWS サインアッププロセスが完了すると、から確認メールが送信されます。<https://aws.amazon.com/> の [マイアカウント] をクリックして、いつでもアカウントの現在のアクティビティを表示し、アカウントを管理することができます。

管理アクセスを持つユーザーを作成する

にサインアップしたら AWS アカウント、日常的なタスクにルートユーザーを使用しないように AWS アカウントのルートユーザー、を保護し AWS IAM アイデンティティセンター、を有効にして管理ユーザーを作成します。

を保護する AWS アカウントのルートユーザー

1. ルートユーザーを選択し、AWS アカウント E メールアドレスを入力して、アカウント所有者[AWS マネジメントコンソール](#)としてにサインインします。次のページでパスワードを入力します。

ルートユーザーを使用してサインインする方法については、AWS サインイン ユーザーガイドの[ルートユーザーとしてサインインする](#)を参照してください。

2. ルートユーザーの多要素認証 (MFA) を有効にします。

手順については、IAM [ユーザーガイドの AWS アカウント「ルートユーザー \(コンソール\) の仮想 MFA デバイス](#)を有効にする」を参照してください。

管理アクセスを持つユーザーを作成する

1. IAM アイデンティティセンターを有効にします。

手順については、「AWS IAM アイデンティティセンター ユーザーガイド」の「[AWS IAM アイデンティティセンターの有効化](#)」を参照してください。

2. IAM アイデンティティセンターで、ユーザーに管理アクセスを付与します。

を ID ソース IAM アイデンティティセンターディレクトリとして使用する方法のチュートリアルについては、AWS IAM アイデンティティセンター「ユーザーガイド」の「[デフォルトを使用してユーザーアクセスを設定する IAM アイデンティティセンターディレクトリ](#)」を参照してください。

管理アクセス権を持つユーザーとしてサインインする

- IAM アイデンティティセンターのユーザーとしてサインインするには、IAM アイデンティティセンターのユーザーの作成時に E メールアドレスに送信されたサインイン URL を使用します。

IAM Identity Center ユーザーを使用してサインインする方法については、AWS サインイン「[ユーザーガイド](#)」の [AWS 「アクセスポータルにサインインする」を参照してください。](#)

追加のユーザーにアクセス権を割り当てる

1. IAM アイデンティティセンターで、最小特権のアクセス許可を適用するというベストプラクティスに従ったアクセス許可セットを作成します。

手順については、「AWS IAM アイデンティティセンター ユーザーガイド」の [「権限設定を作成する」](#) を参照してください。

2. グループにユーザーを割り当て、そのグループにシングルサインオンアクセス権を割り当てます。

手順については、「AWS IAM アイデンティティセンター ユーザーガイド」の [「Add groups」](#) を参照してください。

プログラムによるアクセスの取得

のユーザー サインイン 認証情報に加えて AWS マネジメントコンソール、AWS Command Line Interface () を使用する場合など、プログラムによるアクセス用の認証情報が必要です AWS CLI。プログラム認証情報は、2 つの部分からなるアクセスキーのセットで構成されます。アクセスキーを生成するには、次のいずれかの方法を使用します。

- 方法 1 – IAM アイデンティティセンターで作成された管理ユーザーを使用している場合は、[AWS CLI 「アクセス用の IAM ロール認証情報の取得」](#) を参照して、AWS リソースへの短期アクセス用の一時的なセキュリティ認証情報を生成します。これらの手順に従うときは、管理ユーザー名とパスワード (ルートユーザーではない) を使用して、アカウントの AWS アクセスポータル URL からサインインしていることを確認してください。
- 方法 2 – 既存の IAM ユーザーを使用していて、IAM Identity Center の使用に移行していない場合は、[「IAM ユーザーのアクセスキーの管理 \(コンソール\)」](#) を参照して、ユーザーの長期的な認証情報を生成します。

Note

ベストプラクティスとして、長期的なアクセスキーの代わりに一時的な認証情報を使用します。一時的な認証情報には、アクセスキー ID、シークレットアクセスキー、および認証情報の失効を示すセキュリティトークンが含まれています。詳細については、『』の[AWS 「アクセキーを管理するためのベストプラクティス」](#)を参照してくださいAWS 全般のリファレンス。

Amazon GameLift Streams Web SDK をダウンロードする

コンソール内のストリーミングエクスペリエンスを使用すると、追加のマテリアルなしで開始できます。これは、追加のインフラストラクチャを設定することなく、Amazon GameLift Streams でのアプリケーションのパフォーマンスを評価できるため、開始点としてお勧めします。詳細については、「[Amazon GameLift Streams の開始方法](#)」に進んでください。

独自の Amazon GameLift Streams 統合を構築する準備ができたら、Amazon GameLift Streams Web SDK をダウンロードします。Amazon GameLift Streams Web SDK は、[入門製品ページ](#)のリソースセクションにあります。Amazon GameLift Streams は、ウェブアプリケーションに統合されるように設計されています。JavaScript ベースの Web SDK を統合して、ウェブサイトまたはブラウザベースのアプリケーションからのストリーミングを設定する必要があります。ダウンロードには、Amazon GameLift Streams サービスを使用するサンプルウェブサーバーと、ストリームに接続するためのサンプルウェブクライアントも含まれています。

独自の Amazon GameLift Streams ソリューションの設定の詳細については、「」を参照してください[Amazon GameLift Streams バックエンドサービスとウェブクライアント](#)。

のダウンロード AWS CLI

コンテンツで Amazon GameLift Streams を使用するには、AWS Command Line Interface () を取得することをお勧めしますAWS CLI。AWS CLI は、ターミナルプログラムからコマンドを実行して同等の AWS SDK 機能を提供するオープンソースツールです。

1. オペレーティングシステム AWS CLI 用の の最新バージョンをダウンロードしてインストールします。AWS Command Line Interface ユーザーガイドのこれらの[インストール手順](#)を参照してください。

2. 「[のセットアップ AWS CLI](#)」の説明に従って、ユーザーアクセス認証情報やその他の設定を使用してツールを設定します。この設定では、すべてのコマンドで認証情報やその他の設定を明示的に指定する必要はありません。
3. 次のコマンドを使用してインストールを確認し、使用可能な Amazon GameLift Streams コマンドのリストを取得します。

```
aws gameliftstreams help
```

請求アラートを設定する

ストリームグループには、1秒あたりのアクティブなストリーム容量あたりのコストが発生します。コストと使用量が予算内に収まるようにするには、「」を参照してください[請求アラートを作成して、使用状況をモニタリングする](#)。

Amazon GameLift Streams の開始方法

このセクションでは、Amazon GameLift Streams を通じてアプリケーションとゲームのストリーミングを正常に開始するのに役立ちます。このセクションのトピックでは、Amazon GameLift Streams へのアプリケーションのアップロードから、ストリームでのコンテンツの動作のテストまで、end-to-endのプロセスについて説明します。また、パフォーマンスとコストを最適化するための適切なランタイムとストリームクラス設定の選択など、ストリーミングの準備に役立つ重要なステップについても説明します。

トピック

- [Amazon GameLift Streams での設定の選択](#)
- [設定オプション](#)
- [Amazon GameLift Streams で最初のストリームを開始する](#)

Amazon GameLift Streams での設定の選択

このガイドは、Amazon GameLift Streams を介してアプリケーションやゲームをストリーミングするための最適なランタイム環境と設定を選択するのに役立ちます。設定は、コンテンツのパフォーマンスと Amazon GameLift Streams での実行に関連するコストに直接影響します。さまざまなアプリケーションとグラフィカルな忠実度をサポートするには、いくつかのオプションがあります。

設定オプションの完全なリストは、「」で確認できます[設定オプション](#)。

以下の主要な用語は、これらの設定オプションがどのように連携するかを理解するのに役立ちます。

- ランタイムは、Amazon GameLift Streams でアプリケーションを実行する基盤となるオペレーティングシステムとソフトウェア環境を指します。ランタイム環境の主なオプションは、Windows、Linux、および Proton です。
- ストリームクラスは、Amazon GameLift Streams で使用できるさまざまなリソース設定を表し、オペレーティングシステム、CPU、GPU、RAM、その他の仕様によって異なります。ストリームクラスは、ストリームセッションに割り当てられたハードウェアリソースとテナンシーモデル(1つの仮想マシンで実行できる同時ストリームの数)の両方を定義するストリームグループの設定オプションです。
- マルチテナントにより、複数のユーザーが同じ基盤となるハードウェアリソースを共有できます。これは、最大のハードウェア機能を必要としないアプリケーションにとって費用対効果の高い

オプションです。マルチテナンシーを持つストリームクラスは、1つのリソースのコストで複数のストリームをホストできます。「高」ストリームクラスは 1:2 テナンシーで、「Ultra」ストリームクラスは 1 テナンシーです。

Amazon GameLift Streams の設定時に、選択したランタイム環境によって、互換性があり使用可能な特定のストリームクラスオプションが決まります。Amazon GameLift Streams のパフォーマンスとコスト効率を最適化するには、アプリケーションの要件を適切なランタイム環境とストリームクラスと一致させることが重要です。

ストリーミングのコストは、ストリームクラスによって異なります。コストの詳細なリストについては、Amazon GameLift Streams の[料金ページ](#)を参照してください。

開始点

アプリケーションによっては、ストリーミングを開始するための出発点として最適です。後で、他の設定オプションを調べてコストを最適化できます。

Windows アプリケーションの場合

Microsoft Windows アプリケーションには、Microsoft Windows Server 2022 Base ランタイム環境を使用することをお勧めします。このランタイムには、NVIDIA ベースの gen6n_pro_win2022 、、、gen5n_win2022 および gen4n_win2022 ストリームクラス gen6n_ultra_win2022 である 4 つのハードウェア設定を使用できます。この環境では、Amazon GameLift Streams は DirectX 11 または DirectX 12 を使用するゲームやその他の 3D アプリケーション、および Unity 2022.3、Unreal Engine 4.27、Unreal Engine 5 などのゲームエンジンをまでサポートしています。ストリーミングは IPv4 と IPv6 の両方でサポートされています。

このランタイム環境クラスとストリームクラスの組み合わせは、Windows ベースのコンテンツに対して最高の互換性と最高のパフォーマンスを備えた、予測可能で適切にサポートされた設定を提供します。

Linux アプリケーションの場合

Linux でネイティブに実行するように構築されたアプリケーションには、Ubuntu 22.04 LTS ランタイム環境を使用します。パフォーマンスを最適化するには、Pro または Ultra ストリームクラス (gen6n_pro、、、または gen4n_ultra) gen6n_ultra gen5n_ultra のいずれかを選択します。コストを最適化するには、マルチテナンシーをサポートする Small、Medium、または High ストリームクラス (gen6n_high、gen6n_medium、、、または gen4n_high) gen6n_small

gen5n_high のいずれかを選択します。これは、複数の同時ストリームセッションが同じコンピューティングリソースを共有する費用対効果の高いオプションです。

⚠️ Important

Amazon GameLift Streams の Linux ランタイムは、IPv6 経由のストリーミングをサポートしていません。クライアントは IPv4 経由でアプリケーションをストリーミングする必要があります。

コストの最適化

開始点のレコメンデーションは始めるのに最適ですが、優れたパフォーマンスを維持しながらコストを最適化するために、他の設定オプションを検討することをお勧めします。

Proton ランタイム環境を使用する

多くの Windows アプリケーションは Proton ランタイム環境で実行できます。Proton は、Linux で実行されるゲーム最適化の互換性レイヤーです。このランタイムのストリームクラスオプションには、NVIDIA ハードウェアで実行されている強力な GPU リソースが含まれており、DirectX 11 と Proton 8.0-5、DirectX 12 以降をサポートしています。このオプションの詳細については、[Proton Wiki](#) を参照してください。Proton でのアプリケーションの実行を検討する場合は、Proton 9.0-2 を使用してテストを開始することをお勧めします。

⚠️ Important

Amazon GameLift Streams の Proton ランタイムは、IPv6 経由のストリーミングをサポートしていません。クライアントは IPv4 経由でアプリケーションをストリーミングする必要があります。

⚠️ Important

Proton ランタイム環境における Windows アプリケーションの互換性は、特定のアプリケーション要件によって異なります。例えば、Proton 9.0-2 は、Unreal Engine 5 の Proton 8.0-2c よりも優れたサポートを提供します。一般的に、ゲームが新しいほど、必要な Proton の新しいバージョンになります。最適なパフォーマンスを確保するために、ローカル環境で

このランタイムを徹底的にテストすることを強くお勧めします。[Proton トラブルシューティングガイド](#)を参考にしてください。

アプリケーションを Linux にコンパイルする

もう 1 つのコスト削減オプションは、アプリケーションをターゲットにして Linux でネイティブに実行することです。最初にアプリケーションをテストして、アプリケーションの Linux バージョンが必要に応じて動作することを確認します。アプリケーションが Linux で正常に実行される場合は、Linux アプリケーションの Amazon GameLift Streams 設定オプションに従います。

Unreal Engine アプリケーションを Linux にクロスコンパイルする方法については、Unreal Engine の開発者ガイドの [「Cross-Compile Toolchain」](#) セクションを参照してください。

設定の決定

最適なランタイムとストリームクラスの設定を決定するには、次の重要な質問を検討してください。

1. アプリケーションまたはゲームはどのプラットフォーム用に構築されていますか? Windows アプリケーションがある場合、Windows ランタイム環境は最も簡単にセットアップできます。アプリケーションが Linux 用に構築されている場合、Linux ランタイム環境が最も簡単です。Windows アプリケーションのストリーミングコストを削減するには、Proton ランタイム環境を調べるか、アプリケーションを Linux にコンパイルします。
2. ユースケースでは、パフォーマンスとコストはどの程度重要ですか? Windows ランタイム環境は最高のパフォーマンスを提供する可能性がありますが、実行コストが高くなる可能性があります。これに対して、Proton ランタイム環境はコスト効率が高くなりますが、パフォーマンスがわずかに低下したり、互換性の問題が発生する可能性があります。これは、Windows ベースのアプリケーションが、利用可能な Proton ランタイムでまだ完全にサポートされていない特定の機能を必要とする可能性があるためです。その結果、Proton 環境でアプリケーションを実行するときに、機能的またはグラフィカルな違いが生じる可能性があります。パフォーマンスとコストのトレードオフを評価するために、さまざまなランタイム環境とストリームクラスでアプリケーションをテストすることをお勧めします。ランタイム環境オプションの完全なリストについては、「」を参照してください[ランタイム環境](#)。
3. アプリケーションのグラフィカル要件は何ですか? アプリケーションのグラフィカル要件は、どのストリームクラス設定が最も適切かを判断するのに役立ちます。アプリケーションが高性能 GPUs を必要とする場合は、ビデオメモリ (VRAM) とシステムメモリ (RAM) の量が多いストリームクラスを使用することを検討する必要があります。例えば、gen5n および gen6n ストリームク

ラスは、gen4n ストリームクラスと比較して、グラフィックスを多用するアプリケーションのパフォーマンスが最大 3 倍向上します。アプリケーションで最大 GPU および CPU リソースが必要な場合は、「pro」ストリームクラスを検討する必要があります。逆に、アプリケーションがより低いグラフィカル忠実度で効果的に動作できる場合、マルチテナンシーをサポートするストリームクラス（「small」、「medium」、または「high」のいずれかのストリームクラス）を使用することでコストを削減できます。これにより、複数のユーザーが同じ基盤となるハードウェアリソースを共有できます。ストリームクラスオプションの詳細については、「」を参照してください[ストリームクラス](#)。

4. セットアップにどれだけの労力を費やしますか？ アプリケーションをセットアップする最も簡単な方法は、Windows または Linux ランタイムを使用してネイティブに実行することです。アプリケーション out-of-the-box 互換性がある可能性が高いためです。対照的に、Proton ランタイム環境では、ニーズに最適な Proton 設定を特定するために、より多くの実践的なテストが必要になります。ランタイム環境オプションを決定するときは、セットアップとテストプロセスに割り当てる時間とリソースを考慮してください。
5. さまざまなランタイム環境とストリームクラスでアプリケーションをテストしましたか？ さまざまなランタイム環境とストリームクラスでコンテンツをテストして、その動作を確認することをお勧めします。これにより、安定性、グラフィック品質、機能、入力応答性などの要因に基づいて最適なフィットを判断できます。

設定の選択が次のステップにどのように影響するか

選択した設定は、ストリーミング環境のセットアップの次のフェーズに直接影響します。具体的には次のとおりです。

- Amazon GameLift Streams アプリケーションの作成: ゲームまたはアプリケーションを Amazon GameLift Streams にアップロードするときは、使用するランタイム環境を指定する必要があります。この選択により、使用できるストリームグループのタイプが決まります。
- ストリームグループへのリンク: 既存のストリームグループがある場合は、ランタイム環境の選択がそのグループの設定と一致する必要があります。たとえば、Windows ランタイムを選択した場合、アプリケーションを Windows アプリケーション用に設定されたストリームグループにのみリンクできます。
- ストリームグループの作成: 新しいストリームグループを作成するときは、選択したランタイムと互換性のあるストリームクラスを選択する必要があります。選択したストリームクラスは、アプリケーションに必要なグラフィックス要件とコンピューティング能力と一致する必要があります。

選択した構成設定がこれらの後続のステップにどのように影響するかを理解することで、ストリーミングの全体的な実装をより適切に計画し、スムーズな統合プロセスを確保できます。

次の手順

選択した設定に応じて、ストリーミング用にアプリケーションをセットアップするために使用できる方法がいくつかあります。

Windows または Linux ランタイムを選択した場合

Windows または Linux ランタイムの場合、次のステップは Amazon GameLift Streams でストリーミングを設定し、ストリームをテストすることです。詳細については、「[Amazon GameLift Streams で最初のストリームを開始する](#)」に進んでください。

Proton の使用を検討している場合

アプリケーションの Proton との互換性は、アプリケーションの特定の要件によって異なります。そのため、Amazon GameLift Streams に取り込む前に、さまざまな Proton バージョンでアプリケーションをテストすることをお勧めします。これにより、ニーズに最適なパフォーマンスと互換性を提供する Proton のセットアップを特定できます。Amazon GameLift Streams の外部でテストすることで、アプリケーションのパフォーマンスと機能を検証し、ランタイムに固有の問題をデバッグできます。詳細については、「[Proton for Amazon GameLift Streams との互換性のテストとトラブルシューティング](#)」を参照してください。

特定の Proton 設定を選択すると、Amazon GameLift Streams でストリーミングを設定する準備が整います。詳細については、「」を参照してください[Amazon GameLift Streams で最初のストリームを開始する](#)。

設定オプション

ランタイム環境

ランタイムは、Amazon GameLift Streams でアプリケーションを実行する基盤となるオペレーティングシステムとソフトウェア環境を指します。主なランタイムオプションは、Windows、Linux、および Proton です。開始方法ウォークスルー[ステップ 2: Amazon GameLift Streams 用にアプリケーションを設定する](#)のでランタイム環境を指定します。

Proton は、多くの Windows アプリケーションを Linux ベースの環境で実行できるようにする互換性レイヤーです。Proton を使用する予定がある場合は、ローカルマシンでアプリケーションがどのよ

うに実行されるかをテストすることをお勧めします。詳細については、[Proton for Amazon GameLift Streams との互換性のテストとトラブルシューティング](#) を参照してください。

実行時間	説明
Microsoft Windows Server 2022 Base	Windows アプリケーションと互換性があります。ストリームセッションでの IPv4 と IPv6 の使用をサポートします。
Ubuntu 22.04 LTS	Linux アプリケーションと互換性があります。ストリームセッションでの IPv6 の使用はサポートされていません。
Proton 9.0-2	Windows アプリケーションと互換性があります。Proton experimental_9.0 ブランチに基づいています。Proton との互換性のテストを開始するための推奨バージョン。ストリームセッションでの IPv6 の使用はサポートされていません。
Proton 8.0-5	Windows アプリケーションと互換性があります。Proton experimental_8.0 ブランチに基づいています。ストリームセッションでの IPv6 の使用はサポートされていません。
Proton 8.0-2c	Windows アプリケーションと互換性があります。Proton experimental_8.0 ブランチに基づいています。ストリームセッションでの IPv6 の使用はサポートされていません。

制限事項

ゲームパッドのサポートは Ubuntu 22.04 LTS では利用できません。その他のランタイム環境は、エンドユーザーのオペレーティングシステムとブラウザに応じてゲームパッドをサポートします。詳細については、「[サポートされているブラウザと入力](#)」を参照してください。

ストリームクラス

ストリームクラスは、Amazon GameLift Streams で使用できるさまざまなリソース設定を表し、CPU、GPU、RAM、およびその他の仕様によって異なります。ストリームクラスは、ストリームセッションに割り当てられたハードウェアリソースとテナンシーモデル (1 つの仮想マシンで実行できる同時ストリームの数) の両方を定義するストリームグループの設定オプションです。入門ワークフロー [ステップ 3: Amazon GameLift Streams がアプリケーションをストリーミングする方法を管理する](#) のでストリームクラスを指定します。

ストリームクラス	Amazon EC2 設定	説明
gen6n_pro_win2022	g6.4xlarge Amazon EC2 インスタンスでの Windows ランタイム	<p>(NVIDIA、pro) 最大リソースを必要とする非常に複雑な 3D シーンのアプリケーションをサポートします。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 をサポートします。5.6、32、64 ビットアプリケーションまでの Unreal Engine バージョン、およびチート防止テクノロジーと互換性があります。NVIDIA L4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 16。 RAM: 64 GB。 VRAM: 24 GB。</p> <p>テナント: 最大 1 つの同時ストリームセッションをサポートします。</p>
gen6n_pro	g6.4xlarge Amazon EC2 インスタンスでの Linux ランタイム	<p>(NVIDIA、pro) 最大リソースを必要とする非常に複雑な 3D シーンのアプリケーションをサポートします。NVIDIA L4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 16。 RAM: 64 GB。 VRAM: 24 GB。</p> <p>テナント: 最大 1 つの同時ストリームセッションをサポートします。</p>
gen6n_ultra_win2022	g6.2xlarge Amazon EC2 インスタンスでの Windows ランタイム	<p>(NVIDIA、超) 3D シーンの複雑さが高いアプリケーションをサポートします。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 をサポートします。5.6、32、64 ビットアプリケーションまでの Unreal Engine バージョン、およびチート防止テクノロジーと互</p>

ストリームクラス	Amazon EC2 設定	説明
		<p>換性があります。NVIDIA L4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。RAM: 32 GB。VRAM: 24 GB。</p> <p>テナンシー: 最大 1 つの同時ストリームセッションをサポートします。</p>
gen6n_ultra	g6.2xlarge Amazon EC2 インスタンスでの Linux ランタイム	<p>(NVIDIA、超) 3D シーンの複雑さが高いアプリケーションをサポートします。NVIDIA L4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。RAM: 32 GB。VRAM: 24 GB。</p> <p>テナンシー: 最大 1 つの同時ストリームセッションをサポートします。</p>
gen6n_high	2:1 テナンシーの g6.2xlarge Amazon EC2 インスタンスでの Linux ランタイム	<p>(NVIDIA、高) moderate-to-high の 3D シーンの複雑さを持つアプリケーションをサポートします。NVIDIA L4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 4。RAM: 16 GB。VRAM: 12 GB。</p> <p>テナンシー: 最大 2 つの同時ストリームセッションをサポートします。</p>

ストリームクラス	Amazon EC2 設定	説明
gen6n_medium	4:1 テナンシーの g6.2xlarge Amazon EC2 インスタンスでの Linux ランタイム	(NVIDIA、中) 3D シーンの複雑さが中程度のアプリケーションをサポートします。 NVIDIA L4 Tensor Core GPU を使用します。 アプリケーションあたりのリソース: vCPUs: 2。 RAM: 8 GB。 VRAM: 6 GB。 テナンシー: 最大 4 つの同時ストリームセッションをサポートします。
gen6n_small	12:1 テナンシーの g6.4xlarge Amazon EC2 インスタンスでの Linux ランタイム	(NVIDIA、スマート) 軽量の 3D シーンの複雑さと CPU 使用率が低いアプリケーションをサポートします。 NVIDIA L4 Tensor Core GPU を使用します。 アプリケーションあたりのリソース: vCPUs: 1。 RAM: 4 GB。 VRAM: 2 GB。 テナンシー: 最大 12 の同時ストリームセッションをサポートします。

ストリームクラス	Amazon EC2 設定	説明
gen5n_win_2022	g5.2xlarge Amazon EC2 インスタンスでの Windows ランタイム	<p>(NVIDIA、超) 非常に複雑な 3D シーンのアプリケーションをサポートします。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 と DirectX 11 をサポートします。Unreal Engine をバージョン 5.6、32、64 ビットアプリケーション、およびチート防止テクノロジーでサポートします。NVIDIA A10G Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。RAM: 32 GB。VRAM: 24 GB。</p> <p>テナンシー: 1 つの同時ストリームセッションをサポートします。</p>
gen5n_ultra	g5.2xlarge Amazon EC2 インスタンスでの Linux ランタイム	<p>(NVIDIA、超) 非常に複雑な 3D シーンのアプリケーションをサポートします。NVIDIA A10G Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。RAM: 32 GB。VRAM: 24 GB。</p> <p>テナンシー: 1 つの同時ストリームセッションをサポートします。</p>
gen5n_high	2:1 テナンシーの g5.2xlarge Amazon EC2 インスタンスでの Linux ランタイム	<p>(NVIDIA、高) moderate-to-high の 3D シーンの複雑さを持つアプリケーションをサポートします。NVIDIA A10G Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 4。RAM: 16 GB。VRAM: 12 GB。</p> <p>テナンシー: 最大 2 つの同時ストリームセッションをサポートします。</p>

ストリームクラス	Amazon EC2 設定	説明
gen4n_win_2022	g4dn.2xlarge Amazon EC2 インスタンスでの Windows ランタイム	<p>(NVIDIA、超) 3D シーンの複雑さが高いアプリケーションをサポートします。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 と DirectX 11 をサポートします。バージョン 5.6、32 および 64 ビットアプリケーション、およびチート防止テクノロジーまでの Unreal Engine をサポートします。NVIDIA T4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。RAM: 32 GB。VRAM: 16 GB。</p> <p>テナント: 1 つの同時ストリームセッションをサポートします。</p>
gen4n_ultra	g4dn.2xlarge Amazon EC2 インスタンスでの Linux ランタイム	<p>(NVIDIA、超) 3D シーンの複雑さが高いアプリケーションをサポートします。NVIDIA T4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。RAM: 32 GB。VRAM: 16 GB。</p> <p>テナント: 1 つの同時ストリームセッションをサポートします。</p>
gen4n_high	2:1 テナントの g4dn.2xlarge Amazon EC2 インスタンスでの Linux ランタイム	<p>(NVIDIA、高) moderate-to-high の 3D シーンの複雑さを持つアプリケーションをサポートします。NVIDIA T4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 4。RAM: 16 GB。VRAM: 8 GB。</p> <p>テナント: 最大 2 つの同時ストリームセッションをサポートします。</p>

ランタイム環境とストリームクラスの互換性

ランタイム環境	互換性のあるストリームクラス
Server	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 gen4n_high
プロトン	gen6n_pro gen6n_ultra gen6n_high gen6n_medium gen6n_small gen5n_ultra

ランタイム環境	互換性のあるストリームクラス
	gen5n_high
	gen4n_ultra
	gen4n_high

Amazon GameLift Streams で最初のストリームを開始する

このチュートリアルでは、Amazon GameLift Streams を使用してアプリケーションまたはゲームをストリーミングする手順について説明します。Amazon GameLift Streams はアプリケーションを実行し、エンドユーザーのウェブブラウザに直接ストリーミングします。ストリーミングするアプリケーションをアップロードして設定する方法と、Amazon GameLift Streams ストリームの管理办法について説明します。最終的には、Amazon GameLift Streams コンソールで直接操作することで、Amazon GameLift Streams でアプリケーションがどのようにストリーミングされるかをテストします。

⚠️ 開始する前に、Amazon GameLift Streams の料金を確認してください。

Amazon GameLift Streams のコストについては、[料金表ページ](#)を参照してください。詳細については、[Amazon GameLift Streams の使用状況と請求の管理](#)を参照してください。

Amazon GameLift Streams の使用には、特に以下の場合にコストが発生します。

- で Amazon GameLift Streams アプリケーションを作成する [ステップ 2: Amazon GameLift Streams 用にアプリケーションを設定する](#)
- でストリームグループを作成する [ステップ 3: Amazon GameLift Streams がアプリケーションをストリーミングする方法を管理する](#)

をスキップしないでください[ステップ 5: クリーンアップする \(スキップしない\)](#)。Amazon GameLift Streams の試用が完了した後に不要な料金が発生しないようにするには、すべてのリソースをクリーンアップする必要があります。

トピック

- [前提条件](#)
- [ステップ 1: アプリケーションを Amazon S3 バケットにアップロードする](#)

- [ステップ 2: Amazon GameLift Streams 用にアプリケーションを設定する](#)
- [ステップ 3: Amazon GameLift Streams がアプリケーションをストリーミングする方法を管理する](#)
- [ステップ 4: Amazon GameLift Streams でストリームをテストする](#)
- [ステップ 5: クリーンアップする \(スキップしない\)](#)

前提条件

チュートリアルを開始する前に、以下のタスクを完了してください。

- AWS アカウントにサインアップし、管理アクセス権を持つユーザーをまだ持っていない場合は作成します。このタスクについては、このガイドの[設定トピック](#)を参照してください。AWS CLI 現時点では、Amazon GameLift Streams Web SDK をダウンロードしたり、[をセットアップ](#)したりする必要はありません。[を使用して](#)次の手順を実行します AWS マネジメントコンソール。
- デジタル著作権管理 (DRM) なしでアプリケーションコンテンツファイルのバージョンを取得します。実行可能ファイルやアセットなど、アプリケーションの実行に必要なファイルをフォルダに収集しますが、フォルダを圧縮しないでください。

ステップ 1: アプリケーションを Amazon S3 バケットにアップロードする

Amazon GameLift Streams は Amazon Simple Storage Service (Amazon S3) を使用してアプリケーションまたはゲームファイルをクラウドに保存し、ストリーミングのためにアクセスします。このステップでは、アプリケーションファイルを Amazon S3 バケットにアップロードします。Amazon S3 コンソールでこのステップを完了します。

Note

Amazon GameLift Streams に必要な Amazon S3 ストレージクラスは、デフォルトの S3 Standard です。S3 Glacier や S3 Intelligent-Tiering によって低頻度アクセスまたはアーカイブアクセスに移動されるオブジェクトなどの他のストレージクラスは、Amazon GameLift Streams ではサポートされていません。

ストレージコストを最適化するには、完了[ステップ 2: Amazon GameLift Streams 用にアプリケーションを設定する](#)してアプリケーションが準備完了ステータスになった後、S3 バケットからアプリケーションを削除できます。

アプリケーションの制限

名前	デフォルト	引き上げ可能	説明
アプリケーションあたりのファイル数	30,000 ファイル	はい*	このアカウントで 1 つのアプリケーションに保持できるファイルの最大数。
単一ファイルサイズ	80 GiB	不可	アプリケーション内の 1 つのファイルの最大サイズ。ギガバイト (GiB) は、 $1024 \times 1024 \times 1024$ バイトに等しいことに注意してください。
アプリケーションサイズ	100 GiB	はい*	このアカウントの Amazon GameLift Streams アプリケーションの最大合計サイズ。ギガバイト (GiB) は、 $1024 \times 1024 \times 1024$ バイトに等しいことに注意してください。

*引き上げをリクエストするには、にサインイン AWS マネジメントコンソールし、Service Quotas コンソールを [Amazon GameLift Streams](#) に開きます。ここでは、Applied account-level quota value 列で現在のクオータを確認し、値を引き上げるリクエストを送信できます。

アプリケーションを Amazon S3 にアップロードするには

1. にサインイン AWS マネジメントコンソールし、<https://console.aws.amazon.com/s3/> で Amazon S3 コンソールを開きます。

- Amazon S3 バケットを作成する。バケット名を入力し、を選択します AWS リージョン。このリージョンは、後で作成するアプリケーションおよびストリームグループと同じである必要があります。Amazon GameLift Streams が利用可能な AWS リージョンのリスト[AWS リージョン Amazon GameLift Streams でサポートされている およびリモートロケーション](#)については、「」を参照してください。残りのフィールドについては、デフォルト設定のままにします。

詳細については、「Amazon Simple Storage Service ユーザーガイド」の「[バケットの作成](#)」を参照してください。

- 新しいバケットを開き、アプリケーションファイルとともにフォルダをアップロードします。

 Warning

アプリケーションファイルを非圧縮フォルダとしてアップロードする必要があります。.zip フォルダをアップロードしないでください。

 Warning

アップロードしたアプリケーションファイルが正しいファイルであり、アプリケーションファイルサイズの制限内であることを確認します。後でファイルを更新する場合は、を繰り返す必要があります。[ステップ 2: Amazon GameLift Streams 用にアプリケーションを設定する](#)これには数分かかる場合があります。

ステップ 2: Amazon GameLift Streams 用にアプリケーションを設定する

Amazon GameLift Streams のアプリケーションとは

Amazon GameLift Streams アプリケーションは、Amazon GameLift Streams インフラストラクチャで実行され、クラウドストリーミングを通じてプレイヤーにゲームプレイ体験を提供するゲームまたはインタラクティブアプリケーションを含むリソースです。アプリケーションは AWS コンピューティングインスタンスで実行され、インターネット経由でプレイヤーのデバイスに直接ストリーミングされるゲームコンテンツをレンダリングするため、プレイヤーがゲームをローカルでダウンロード、インストール、または実行する必要がなくなります。

このステップでは、Amazon GameLift Streams アプリケーションを作成して、Amazon GameLift Streams でストリーミングするアプリケーションを設定します。Amazon GameLift Streams アプリ

ケーションを作成するときは、Amazon S3 バケットにアップロードしたアプリケーションフォルダに Amazon S3 URI を指定し、有効な実行可能ファイルまたはスクリプトファイルへの相対パスを指定します。Amazon GameLift Streams コンソールでこのステップを完了します。

Amazon GameLift Streams コンソールを使用して Amazon GameLift Streams アプリケーションを作成するには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソール](#)を開きます。ファイルセットをアップロードした Amazon S3 バケット AWS リージョンと同じを選択します。詳細については、「[入門ガイド](#)」の「[リージョンの選択](#)」を参照してください。
AWS マネジメントコンソール
2. ナビゲーションバーで、アプリケーションを選択し、アプリケーションの作成を選択します。
3. ランタイム設定で、次のように入力します。
 - ランタイム環境

これは、アプリケーションを実行するランタイム環境です。Amazon GameLift Streams は、Windows、Ubuntu 22.04 LTS、または [Proton](#) のいずれかで実行できます。

このフィールドを作成ワークフローの後に編集することはできません。

次のいずれかのランタイム環境から選択。

- Linux アプリケーションの場合:
 - Ubuntu 22.04 LTS (UBUNTU, 22_04_LTS)
- Windows アプリケーションの場合:
 - Microsoft Windows Server 2022 Base (WINDOWS, 2022)
 - Proton 9.0-2 (PROTON, 20250516)
 - Proton 8.0-5 (PROTON, 20241007)
 - Proton 8.0-2c (PROTON, 20230704)

説明を確認し、比較チェックリストを使用して、アプリケーションに最適なランタイム環境を選択します。

4. 全般設定で、次のように入力します。

- a. 説明

人間が読み取れるアプリケーション用のラベルです。この値は一意である必要はありません。ベストプラクティスとして、アプリケーションのわかりやすい説明、名前、またはラベルを使用してください。このフィールドはいつでも編集できます。

b. ベースパス

これは、Amazon S3 バケット内のアプリケーションのルートフォルダへの Amazon S3 URI です。このフォルダおよびサブフォルダには、ビルド実行ファイルとサポートファイルを含める必要があります。

有効な URI は、アプリケーションの実行とストリーミングに必要なすべてのファイルを含むバケットプレフィックスです。例: mygamebuild というバケットに、完全なバージョンのゲームビルドファイルが 3 つ、それぞれ別のフォルダに含まれているとします。フォルダ mygamebuild-EN101 内のビルドをストリーミングします。この例では、URI は s3://amzn-s3-demo-bucket/mygamebuild-EN101 です。

このフィールドを作成ワークフローの後に編集することはできません。

c. 実行可能な起動パス

これは、Amazon GameLift Streams がストリーミングする実行ファイルの Amazon S3 URI です。このファイルは、アプリケーションのルートフォルダに含める必要があります。Windows アプリケーションの場合、ファイルは、.exe、.cmd、または .bat で終わるファイル名を持つ有効な Windows 実行可能ファイルまたはバッチファイルである必要があります。Linux アプリケーションの場合、ファイルは有効な Linux バイナリ実行可能ファイルであるか、シェバン ('#!') で始まる最初のインタープリタ行を含むスクリプトである必要があります。

このフィールドを作成ワークフローの後に編集することはできません。

5. (オプション) アプリケーションログパスに、次のように入力します。

a. アプリケーションログパス

これは、保存するログを含むアプリケーションフォルダまたはファイルへのパス(1つまたは複数)です。各ログパスは、アプリケーションのベースパスを基準にして指定します。この機能を使用すると、各ストリームセッションの最後に、Amazon GameLift Streams は指定されたファイルを、指定された名前の Amazon S3 バケットにコピーします。コピー操作は、アプリケーションフォルダのサブフォルダでは、再帰的に実行されません。

ログ記録を無効にするには、すべてのアプリケーションログパスを削除し、アプリケーションのログ出力先をクリアします。

このフィールドはいつでも編集できます。

b. アプリケーションログの出力

これは、Amazon GameLift Streams がアプリケーションログファイルをコピーする Amazon S3 バケットの URI です。このフィールドは、アプリケーションログパスを指定する場合に必須です。

ログ記録を無効にするには、すべてのアプリケーションログパスを削除し、アプリケーションのログ出力先をクリアします。

このフィールドはいつでも編集できます。

自動でログファイルを保存するには、Amazon GameLift Streams に S3 バケットへの保存アクセス許可を与える必要があります。ログ記録用のバケットを Amazon GameLift Streams に作成させると、作成時にアクセス許可ポリシーが自動的に適用されます。独自のバケットを指定する場合は、アクセス許可ポリシーを自分で適用する必要があります。

バケットアクセス許可ポリシーテンプレート

次のポリシーコードをコピーして、アプリケーションログに使用するバケットに適用します。必ず amzn-s3-demo-bucket を既存の S3 バケットの名前に置き換えてください。

```
{
    "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. (オプション) タグで、このアプリケーションにタグを割り当てます。

タグは、AWS リソースの整理に役立つラベルです。詳細については、「[AWS リソースのタグ付け](#)」を参照してください。

たとえば、アプリケーションバージョンを追跡するには、などのタグを使用します application-version : my-game-1121。

7. [アプリケーションを作成] を選択します。

Amazon GameLift Streams は、アプリケーションの準備に数分かかります。アプリケーションページで、新しいアプリケーションは処理中ステータスです。アプリケーションが準備完了ステータスになったら、次のステップ「」に進むことができます [ステップ 3: Amazon GameLift Streams がアプリケーションをストリーミングする方法を管理する](#)。

リクエストがエラーを返した場合、またはアプリケーションが作成されたがエラーステータスである場合は、Amazon S3 と Amazon GameLift Streams の両方へのアクセスを含むユーザー認証情報を使用していることを確認してください。

 Note

アプリケーションのステータスが準備完了の場合、新しいアプリケーションに影響を与えることなく、Amazon S3 バケット内のアプリケーションファイルを安全に削除できます。これは、ストレージコストの最適化にも役立ちます。詳細については、「[アプリケーションの削除](#)」を参照してください。

詳細については、[Amazon GameLift Streams でアプリケーションを準備する](#) を参照してください。

ステップ 3: Amazon GameLift Streams がアプリケーションをストリーミングする方法を管理する

ストリームグループとは

ストリームグループを使用して、Amazon GameLift Streams がアプリケーションをストリーミングする方法を管理します。ストリームグループは、Amazon GameLift Streams がアプリケーションをエンドユーザーにストリーミングするために使用するコンピューティングリソースのコレクションです。ストリームグループを作成するときは、ゲームを実行するハードウェア設定 (CPU、GPU、RAM) (ストリームクラスと呼ばれる)、ゲームが実行できる地理的場所、各場所で同時に実行できるストリームの数 (ストリーム容量と呼ばれる) を指定します。ストリームグループの作成時にアプリケーションをリンクすることも、後で待機することもできますが、ストリームグループからストリーミングする前に少なくとも 1 つのアプリケーションをリンクする必要があります。ストリームグループが作成されると、Amazon GameLift Streams はストリーム容量を割り当てた場所にコンピューティングリソースを割り当てます。この時点で、ストリームグループに追加のアプリケーションを関連付けて、どのアプリケーションをストリーミングするかを選択することもできます。

アプリケーションの準備ができたら、次に Amazon GameLift Streams がストリーミングするためのコンピューティングリソースが必要になります。このステップでは、ストリームグループを作成して、Amazon GameLift Streams がアプリケーションをストリーミングする方法を管理します。Amazon GameLift Streams コンソールでこのステップを完了します。

Amazon GameLift Streams コンソールでストリームグループを作成するには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソール](#)を開きます。ストリームグループ AWS リージョン を作成する を選択します。このリージョンは、ストリームグループでストリーミングするアプリケーションのリージョンと同じである必要があります。詳細については、「[入門ガイド](#)」の「[リージョンの選択](#)」を参照してください。AWS マネジメントコンソール
2. 作成ワークフローを開くには、ナビゲーションペインでストリームグループを選択し、ストリームグループの作成を選択します。
3. ストリームグループの定義で、次のように入力します。
 - a. 説明

人間が読み取れるストリームグループ用のラベル。この値は一意である必要はありません。ベストプラクティスとして、ストリームグループのわかりやすい説明、名前、またはラベルを使用してください。このフィールドはいつでも編集できます。

b. タグ

タグは、AWS リソースの整理に役立つラベルです。詳細については、「[AWS リソースのタグ付け](#)」を参照してください。

4. Select stream class で、ストリームグループのストリームクラスを選択します。

- ストリームクラスオプション

アプリケーションを実行およびストリーミングするコンピューティングリソースのタイプ。この選択は、ストリーミングエクスペリエンスの品質とコストに影響します。ストリームグループごとに指定できるストリームクラスは 1 つだけです。アプリケーションに最適なクラスを選択します。

ストリームクラス	説明
gen6n_pro _win2022	(NVIDIA、pro) 最大リソースを必要とする非常に複雑な 3D シーンのアプリケーションをサポートします。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 をサポートします。5.6、32、64 ビットアプリケーションまでの Unreal Engine バージョン、およびチート防止テクノロジーと互換性があります。NVIDIA L4 Tensor Core GPU を使用します。 アプリケーションあたりのリソース: vCPUs: 16。RAM: 64 GB。VRAM: 24 GB。 テナント: 最大 1 つの同時ストリームセッションをサポートします。
gen6n_pro	(NVIDIA、pro) 最大リソースを必要とする非常に複雑な 3D シーンのアプリケーションをサポートします。NVIDIA L4 Tensor Core GPU を使用します。 アプリケーションあたりのリソース: vCPUs: 16。RAM: 64 GB。VRAM: 24 GB。

ストリームクラス	説明
	<p>テナンシー: 最大 1 つの同時ストリームセッションをサポートします。</p>
gen6n_ult ra_win2022	<p>(NVIDIA、超) 3D シーンの複雑さが高いアプリケーションをサポートします。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 をサポートします。5.6、32、64 ビットアプリケーションまでの Unreal Engine バージョン、およびチート防止テクノロジーと互換性があります。NVIDIA L4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。RAM: 32 GB。VRAM: 24 GB。</p> <p>テナンシー: 最大 1 つの同時ストリームセッションをサポートします。</p>
gen6n_ultra	<p>(NVIDIA、超) 3D シーンの複雑さが高いアプリケーションをサポートします。NVIDIA L4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。RAM: 32 GB。VRAM: 24 GB。</p> <p>テナンシー: 最大 1 つの同時ストリームセッションをサポートします。</p>
gen6n_high	<p>(NVIDIA、高) moderate-to-high の 3D シーンの複雑さを持つアプリケーションをサポートします。NVIDIA L4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 4。RAM: 16 GB。VRAM: 12 GB。</p> <p>テナンシー: 最大 2 つの同時ストリームセッションをサポートします。</p>

ストリームクラス	説明
gen6n_medium	<p>(NVIDIA、中) 3D シーンの複雑さが中程度のアプリケーションをサポートします。NVIDIA L4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 2。 RAM: 8 GB。 VRAM: 6 GB。</p> <p>テナント: 最大 4 つの同時ストリームセッションをサポートします。</p>
gen6n_small	<p>(NVIDIA、スマート) 軽量の 3D シーンの複雑さと CPU 使用率が低いアプリケーションをサポートします。NVIDIA L4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 1。 RAM: 4 GB。 VRAM: 2 GB。</p> <p>テナント: 最大 12 の同時ストリームセッションをサポートします。</p>
gen5n_win2022	<p>(NVIDIA、超) 非常に複雑な 3D シーンのアプリケーションをサポートします。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 と DirectX 11 をサポートします。Unreal Engine をバージョン 5.6、32、64 ビットアプリケーション、およびチート防止テクノロジーでサポートします。NVIDIA A10G Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。 RAM: 32 GB。 VRAM: 24 GB。</p> <p>テナント: 1 つの同時ストリームセッションをサポートします。</p>

ストリームクラス	説明
gen5n_high	<p>(NVIDIA、高) moderate-to-highの 3D シーンの複雑さを持つアプリケーションをサポートします。NVIDIA A10G Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 4。RAM: 16 GB。VRAM: 12 GB。</p> <p>テナンシー: 最大 2 つの同時ストリームセッションをサポートします。</p>
gen5n_ultra	<p>(NVIDIA、超) 非常に複雑な 3D シーンのアプリケーションをサポートします。NVIDIA A10G Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。RAM: 32 GB。VRAM: 24 GB。</p> <p>テナンシー: 1 つの同時ストリームセッションをサポートします。</p>
gen4n_win2022	<p>(NVIDIA、超) 3D シーンの複雑さが高いアプリケーションをサポートします。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 と DirectX 11 をサポートします。Unreal Engine をバージョン 5.6、32、64 ビットアプリケーション、およびチート防止テクノロジーでサポートします。NVIDIA T4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。RAM: 32 GB。VRAM: 16 GB。</p> <p>テナンシー: 1 つの同時ストリームセッションをサポートします。</p>

ストリームクラス	説明
gen4n_high	<p>(NVIDIA、高) moderate-to-high の 3D シーンの複雑さを持つアプリケーションをサポートします。NVIDIA T4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 4。RAM: 16 GB。VRAM: 8 GB。</p> <p>テナンシー: 最大 2 つの同時ストリームセッションをサポートします。</p>
gen4n_ultra	<p>(NVIDIA、超) 3D シーンの複雑さが高いアプリケーションをサポートします。NVIDIA T4 Tensor Core GPU を使用します。</p> <p>アプリケーションあたりのリソース: vCPUs: 8。RAM: 32 GB。VRAM: 16 GB。</p> <p>テナンシー: 1 つの同時ストリームセッションをサポートします。</p>

続行するには、次へ を選択します。

- Link application で、ストリーミングするアプリケーションを選択するか、「アプリケーションなし」を選択して後で選択します。ストリームグループは、作成後に編集して、アプリケーションを追加または削除できます。Ready ステータスがで、選択したストリームクラスと互換性のあるランタイムを持つアプリケーションのみをリンクできます。デフォルトでは、これらはテーブルに表示される唯一のアプリケーションです。Ready ステータスのすべてのアプリケーションを表示するには、ドロップダウンリストから All runtimes を選択します。

 Note

アプリケーションが表示されない場合は、現在の AWS リージョン 設定を確認してください。アプリケーションをリンクできるのは、同じリージョンにあるストリームグループのみです。

続行するには、次へ を選択します。

6. 「ストリーム設定を構成する」の「場所と容量」で、ストリームグループにアプリケーションをストリーミングする容量がある 1 つ以上の場所を選択します。デフォルトでは、プライマリロケーションと呼ばれるストリームグループを作成するリージョンは、既にストリームグループに追加されており、削除できません。追加する各場所の横にあるチェックボックスをオンにして、場所を追加できます。レイテンシーを短縮し、ストリーミングの品質を向上させるには、ユーザーに近い場所を選択する必要があります。

各ケーションごとに、ストリーミング容量を指定できます。[ストリーム容量] は、一度にアクティブにできる同時実行ストリームの数を表します。各ストリームグループの場所ごとにストリーム容量を設定します。

- **常時オン容量:** ゼロ以外の場合、この設定は、割り当てられた最小ストリーミング容量を示し、サービスに解放されることはありません。この基本レベルの容量については、使用の有無にかかわらず、常にお支払いいただきます。
- **最大容量:** これは、サービスが割り当てることができる最大容量を示します。新しく作成されたストリームの開始には数分かかる場合があります。アイドル状態になると、キャパシティはサービスに解放されます。解放されるまで割り当てられた容量に対して料金が発生します。
- **ターゲットアイドル容量:** これは、将来のアクティビティを想定してサービスが事前に割り当てて保持するアイドル容量を示します。これにより、キャパシティ割り当ての遅延からユーザーを隔離できます。この意図的なアイドル状態で保持されている容量に対して料金が発生します。

いずれかの容量を調整することで、場所のユーザー需要の変化に合わせて、ストリームの合計容量をいつでも増減できます。Amazon GameLift Streams は、常時オンのキャパシティプールにアイドル状態で事前に割り当てられたリソースがあれば、それを使用してストリーミングリクエストを実行します。すべての常時オン容量が使用されている場合、Amazon GameLift Streams はオンデマンド容量で指定された最大数まで追加のコンピューティングリソースをプロビジョニングします。割り当てられた容量がスケールするについて、その変化がストリームグループの総コストに反映されます。

リンクされたアプリケーションは、有効な各場所に自動的にレプリケートされます。リモートロケーションがストリームをホストする前に、アプリケーションはリモートロケーションでのレプリケーションを終了する必要があります。レプリケーションステータスを確認するには、作成後にストリームグループを開き、リンクされたアプリケーションのテーブルのレプリケーションステータス列を参照してください。現在のステータスをクリックすると、追加された各ロケーションのレプリケーションステータスが表示されます。

Note

アプリケーションデータは、このストリームグループのプライマリロケーションを含む、有効なすべてのロケーションに保存されます。ストリームセッションデータは、プライマリロケーションとストリーミングが発生したロケーションの両方に保存されます。

- 「ストリームグループの確認と作成」で、ストリームグループの設定を確認し、必要に応じて変更を加えます。すべて正しい場合は、ストリームグループの作成を選択します。

詳細については、[Amazon GameLift Streams ストリームグループを使用してストリーミングを管理する](#) を参照してください。

ステップ 4: Amazon GameLift Streams でストリームをテストする

ストリームセッションとは

ストリーム自体を指します。これは、Amazon GameLift Streams がサーバーからエンドユーザーに送信するストリームのインスタンスです。ストリームセッションは、ストリームグループが割り当てたコンピューティングリソースまたはストリーム容量で実行されます。ストリームとも呼ばれます。

Amazon GameLift Streams コンソールで直接実行することで、アプリケーションがどのようにストリーミングされるかを確認できます。ストリームを開始すると、Amazon GameLift Streams はストリームグループが割り当てるコンピューティングリソースのいずれかを使用します。そのため、ストリームグループで使用可能な容量が必要です。

Amazon GameLift Streams コンソールでストリームをテストするには

- にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソール](#)を開きます。
- ストリームは、いくつかの方法でテストできます。ストリームグループページまたはテストストリームページから開始し、以下の手順に従います。
 - ストリーミングに使用するストリームグループを選択します。

- b. ストリームグループページから開始する場合は、ストリームのテストを選択します。テストストリームページから開始する場合は、選択を選択します。これにより、選択したストリームグループのストリーム設定のテストページが開きます。
 - c. リンクされたアプリケーションで、アプリケーションを選択します。
 - d. Location で、使用可能な容量がある場所を選択します。
 - e. (オプション) プログラム設定で、起動時にアプリケーションに渡すコマンドライン引数または環境変数を入力します。
 - f. 選択内容を確認し、ストリームのテストを選択します。
3. ストリームがロードされたら、ストリームで次のアクションを実行できます。
 - a. マウス、キーボード、ゲームパッド (テストストリームではサポートされていないマイクを除く) などの入力を接続するには、入力をアタッチを選択します。カーソルをストリームウィンドウに移動すると、マウスが自動的にアタッチされます。
 - b. ストリーミングセッション中に作成されたファイルをセッションの終了時に Amazon S3 バケットにエクスポートするには、ファイルのエクスポートを選択し、バケットの詳細を指定します。エクスポートされたファイルは、「セッション」ページにあります。
 - c. ストリームを全画面で表示するには、全画面表示を選択します。Escape を押して、このアクションを元に戻します。
 4. ストリームを終了するには、セッションの終了を選択します。ストリームが切断されると、ストリーム容量が別のストリームを開始できるようになります。

 Note

Amazon GameLift Streams コンソールのテストストリーム機能はマイクをサポートしていません。

ステップ 5: クリーンアップする (スキップしない)

 不要なコストを回避する

ストリームグループが容量を割り当てた場合、その容量が未使用であってもコストが発生します。不要なコストを回避するには、ストリームグループの容量を必要なサイズにスケールします。開発時には、使用していないときに常時オンの容量をゼロにスケールすることをお

勧めします。詳細については、[Amazon GameLift Streams のコストを管理するためのベストプラクティス](#) を参照してください。

チュートリアルを完了し、アプリケーションをストリーミングする必要がなくなったら、以下の手順に従って Amazon GameLift Streams リソースをクリーンアップします。

Amazon GameLift Streams コンソールを使用してストリームグループを削除するには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソール](#)を開きます。
2. 既存のストリームグループのリストを表示するには、ナビゲーションペインでストリームグループを選択します。
3. 削除するストリームグループの名前を選択します。
4. ストリームグループの詳細ページで、削除を選択します。
5. 削除ダイアログボックスで、削除アクションを確認します。

Amazon GameLift Streams は、コンピューティングリソースのリリースとストリームグループの削除を開始します。この間、ストリームグループは削除ステータスになります。Amazon GameLift Streams がストリームグループを削除すると、そのグループを取得できなくなります。

Amazon GameLift Streams コンソールを使用してアプリケーションを削除するには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソール](#)を開きます。
2. ナビゲーションバーで、アプリケーションを選択して既存のアプリケーションのリストを表示します。削除するアプリケーションを選択します。
3. アプリケーションの詳細ページで、削除を選択します。
4. 削除ダイアログボックスで、削除アクションを確認します。

Amazon GameLift Streams がアプリケーションの削除を開始します。この間、アプリケーションは Deletingステータスになります。Amazon GameLift Streams がアプリケーションを削除すると、アプリケーションを取得できなくなります。

詳細については、[ストリームグループを削除する](#) および [アプリケーションの削除](#) を参照してください。

Amazon GameLift Streams を使用したストリームの管理

このセクションでは、Amazon GameLift Streams でストリーミングする方法について詳しく説明します。ストリーミングリソース (アプリケーションとストリームグループ)、ストリーミングをスケールするプロパティ (ストリームの容量と場所)、ストリーム自体 (ストリームセッション) について説明します。Amazon GameLift Streams コンソールまたは Amazon GameLift Streams CLI コマンドを使用して、Amazon GameLift Streams でストリーミングをセットアップするために必要なすべてのタスクを処理できます。

Amazon GameLift Streams を初めて使用する場合は、[Amazon GameLift Streams で最初のストリームを開始する](#)「」を参照してください。ワークフロー全体について説明します。

トピック

- [主要なコンセプト](#)
- [Amazon GameLift Streams でアプリケーションを準備する](#)
- [Amazon GameLift Streams ストリームグループを使用してストリーミングを管理する](#)
- [マルチアプリケーションストリームグループの概要](#)
- [Amazon GameLift Streams でストリームセッションを開始する](#)
- [ストリームセッションファイルのエクスポート](#)

主要なコンセプト

アプリケーション

Amazon GameLift Streams アプリケーションは、Amazon GameLift Streams インフラストラクチャで実行され、クラウドストリーミングを通じてプレイヤーにゲームプレイ体験を提供するゲームまたはインタラクティブアプリケーションを含むリソースです。アプリケーションは AWS コンピューティングインスタンスで実行され、インターネット経由でプレイヤーのデバイスに直接ストリーミングされるゲームコンテンツをレンダリングするため、プレイヤーがゲームをローカルでダウンロード、インストール、または実行する必要がなくなります。

マルチアプリケーションストリームグループ

複数のアプリケーションにリンクされたストリームグループ。このmany-to-one の関係により、単一のストリームグループで設定したのと同じ設定を使用して、複数のアプリケーションをストリーミングできます。ストリームセッションを開始するときは、リンクされたアプリケーション

を指定します。次に、Amazon GameLift Streams は、このストリームグループで使用可能なストリーム容量を使用してそのアプリケーションをストリーミングします。

マルチロケーションストリームグループ

プライマリロケーション (ストリームグループを作成した AWS リージョン) に加えて、複数のロケーションからアプリケーションとストリームセッションをホストするように設定されたストリームグループ。各ロケーションの容量を管理します。

マルチテナンシー

テナンシーとは、Amazon GameLift Streams の 1 つのコンピューティングリソースでサポートできる同時ストリームの数を指します。マルチテナンシーは、複数のユーザーが同じ基盤となるハードウェアリソースを共有できるようにする機能です。これは、最大のハードウェア機能を必要としないアプリケーションにとって費用対効果の高いオプションです。マルチテナンシーを持つストリームクラスは、1 つのリソースのコストで複数のストリームをホストできます。

「高」ストリームクラスはマルチテナンシーをサポートし、2 つのアプリケーションを 1 つのコンピューティングリソースで同時に実行できますが、「Ultra」ストリームクラスはマルチテナンシーをサポートしていません。

ストリームグループ

ストリームグループを使用して、Amazon GameLift Streams がアプリケーションをストリーミングする方法を管理します。ストリームグループは、Amazon GameLift Streams がアプリケーションをエンドユーザーにストリーミングするために使用するコンピューティングリソースのコレクションです。ストリームグループを作成するときは、ゲームを実行するハードウェア設定 (CPU、GPU、RAM) (ストリームクラスと呼ばれる)、ゲームを実行できる地理的場所、各場所で同時に実行できるストリームの数 (ストリーム容量と呼ばれる) を指定します。ストリームグループの作成時にアプリケーションをリンクすることも、後で待機することができますが、ストリームグループからストリーミングするには、少なくとも 1 つのアプリケーションをリンクする必要があります。ストリームグループが作成されると、Amazon GameLift Streams はストリーム容量を割り当てた場所にコンピューティングリソースを割り当てます。この時点で、ストリームグループに追加のアプリケーションを関連付けて、どのアプリケーションをストリーミングするかを選択することもできます。

ストリーム容量

一度にアクティブにできる同時ストリームの数を表します。各ストリームグループの場所ごとにストリーム容量を設定します。常時オンの容量と最大容量を設定します。最大容量は、可能なストリームの合計を表し、常時オンの容量と必要に応じてプロビジョニングされる追加の容量を組み合わせます。

ストリームセッション

ストリーム自身を指します。これは、Amazon GameLift Streams がサーバーからエンドユーザーに送信するストリームのインスタンスです。ストリームセッションは、ストリームグループが割り当てたコンピューティングリソースまたはストリーム容量で実行されます。ストリームとも呼ばれれます。

Amazon GameLift Streams でアプリケーションを準備する

Amazon GameLift Streams でストリーミングを設定するには、まずストリーミングするゲームまたは他のアプリケーションをアップロードし、Amazon GameLift Streams 内でアプリケーションリソースを設定してゲームに関するメタデータを定義します。Amazon GameLift Streams アプリケーションは、アップロードしたファイル（実行可能ファイルとサポートファイル）と、ストリーミング時に実行する実行可能ファイルを Amazon GameLift Streams に指示する設定で構成されます。

各 Amazon GameLift Streams アプリケーションは、コンテンツの 1 つのバージョンを表します。複数のバージョンがある場合は、バージョンごとに個別のアプリケーションを作成する必要があります。アプリケーションを作成した後は、ファイルを更新することはできません。実行可能ファイルまたはサポートファイルを更新する必要がある場合は、新しい Amazon GameLift Streams アプリケーションを作成する必要があります。

アップロードする前に

Amazon GameLift Streams アプリケーションを作成する前に、ゲームが次の制限に従っていることを確認してください。

名前	デフォルト	引き上げ可能	説明
アプリケーションあたりのファイル数	30,000 ファイル	はい*	このアカウントでアプリケーションに保持できるファイルの最大数。

名前	デフォルト	引き上げ可能	説明
単一ファイルサイズ	80 GiB	なし	アプリケーション内の 1 つのファイルの最大サイズ。ギガバイト (GiB) は $1024 \times 1024 \times 1024$ バイトに等しいことに注意してください。
アプリケーションサイズ	100 GiB	はい*	このアカウントの Amazon GameLift Streams アプリケーションの最大合計サイズ。ギガバイト (GiB) は $1024 \times 1024 \times 1024$ バイトに等しいことに注意してください。

*引き上げをリクエストするには、[AWS マネジメントコンソール](#) にサインインし、Service Quotas コンソールから [Amazon GameLift Streams](#) を開きます。ここでは、Applied アカウントレベルの クオータ値列で現在のクオータを確認し、値引き上げのリクエストを送信できます。

Note

時間と労力を節約するには、アップロードする準備ができているファイルがアプリケーションの正しいバージョンであることを確認します。新しいバージョンは後でアップロードできますが、バージョンごとに [アプリケーションの作成](#) ステップを繰り返す必要があります。

Amazon S3 バケットにアプリケーションをアップロードする

Amazon GameLift Streams 用にゲームを準備したので、AWS アカウント内の Amazon Simple Storage Service (Amazon S3) バケットにアップロードします。

Note

Amazon GameLift Streams に必要な Amazon S3 ストレージクラスは、デフォルトの S3 Standard です。S3 Glacier や S3 Intelligent-Tiering によって低頻度アクセスまたはアーカイブアクセスに移動されるオブジェクトなどの他のストレージクラスは、Amazon GameLift Streams ではサポートされていません。

ストレージコストを最適化するには、完了 [アプリケーションの作成](#)してアプリケーションが準備完了ステータスになった後、S3 バケットからアプリケーションを削除できます。

アプリケーションを Amazon S3 にアップロードするには

1. にサインイン AWS マネジメントコンソール し、<https://console.aws.amazon.com/s3/> で Amazon S3 コンソールを開きます。
2. Amazon S3 バケットを作成する。バケット名を入力し、を選択します AWS リージョン。このリージョンは、後で作成するアプリケーションおよびストリームグループと同じである必要があります。Amazon GameLift Streams が利用可能な AWS リージョン のリスト[AWS リージョン Amazon GameLift Streams でサポートされている およびリモートロケーション](#)については、「」を参照してください。残りのフィールドについては、デフォルト設定のままにします。

詳細については、「Amazon Simple Storage Service ユーザーガイド」の「[バケットの作成](#)」を参照してください。

3. 新しいバケットを開き、アプリケーションファイルとともにフォルダをアップロードします。

Warning

アプリケーションファイルを非圧縮フォルダとしてアップロードする必要があります。.zip フォルダをアップロードしないでください。

アプリケーションの作成

Amazon GameLift Streams アプリケーションは、Amazon GameLift Streams インフラストラクチャで実行され、クラウドストリーミングを通じてプレイヤーにゲームプレイ体験を提供するゲームまたはインタラクティブアプリケーションを含むリソースです。アプリケーションは AWS コンピューティングインスタンスで実行され、インターネット経由でプレイヤーのデバイスに直接ストリーミングされるゲームコンテンツをレンダリングするため、プレイヤーがゲームをローカルでダウンロード、インストール、または実行する必要がなくなります。

Amazon GameLift Streams アプリケーションを作成するときは、Amazon S3 バケットにアップロードしたアプリケーションフォルダに Amazon S3 URI を指定し、有効な実行可能ファイルまたはスクリプトファイルへの相対パスを指定します。

Amazon GameLift Streams は、アプリケーションファイルを Amazon S3 バケット内のファイルと同期させません。Amazon GameLift Streams アプリケーションのファイルを更新する場合は、新しい Amazon GameLift Streams アプリケーションを作成する必要があります。

Console

Amazon GameLift Streams コンソールを使用して Amazon GameLift Streams アプリケーションを作成するには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソールを開きます](#)。ファイルセットをアップロードした Amazon S3 バケット AWS リージョンと同じを選択します。詳細については、「[入門ガイド](#)」の「リージョンの選択」を参照してください。AWS マネジメントコンソール
2. ナビゲーションバーで、アプリケーションを選択し、アプリケーションの作成を選択します。
3. ランタイム設定で、次のように入力します。
 - ランタイム環境

これは、アプリケーションを実行するランタイム環境です。Amazon GameLift Streams は、Windows、Ubuntu 22.04 LTS、または [Proton](#) のいずれかで実行できます。

このフィールドを作成ワークフローの後に編集することはできません。

次のいずれかのランタイム環境から選択。

- Linux アプリケーションの場合:

- Ubuntu 22.04 LTS (UBUNTU, 22_04_LTS)
- Windows アプリケーションの場合:
 - Microsoft Windows Server 2022 Base (WINDOWS, 2022)
 - Proton 9.0-2 (PROTON, 20250516)
 - Proton 8.0-5 (PROTON, 20241007)
 - Proton 8.0-2c (PROTON, 20230704)

説明を確認し、比較チェックリストを使用して、アプリケーションに最適なランタイム環境を選択します。

4. 全般設定で、次のように入力します。

a. 説明

人間が読み取れるアプリケーション用のラベルです。この値は一意である必要はありません。ベストプラクティスとして、アプリケーションのわかりやすい説明、名前、またはラベルを使用してください。このフィールドはいつでも編集できます。

b. ベースパス

これは、Amazon S3 バケット内のアプリケーションのルートフォルダへの Amazon S3 URI です。このフォルダおよびサブフォルダには、ビルド実行ファイルとサポートファイルを含める必要があります。

有効な URI は、アプリケーションの実行とストリーミングに必要なすべてのファイルを含むバケットプレフィックスです。例: というバケットmygamebuildには、ゲームビルドファイルの 3 つの完全なバージョンが含まれており、それぞれが別のフォルダにあります。フォルダ mygamebuild-EN101 内のビルドをストリーミングします。この例では、URI は s3://amzn-s3-demo-bucket/mygamebuild-EN101 です。

このフィールドを作成ワークフローの後に編集することはできません。

c. 実行可能な起動パス

これは、Amazon GameLift Streams がストリーミングする実行可能ファイルの Amazon S3 URI です。GameLift このファイルは、アプリケーションのルートフォルダに含める必要があります。Windows アプリケーションの場合、ファイルは .exe、.cmd、または .bat で終わるファイル名を持つ有効な Windows 実行可能ファイルまたはバッチファイルである必要があります。Linux アプリケーションの場合、ファイルは有効な Linux

バイナリ実行可能ファイルであるか、シェバン ('#!') で始まる最初のインタープリタ行を含むスクリプトである必要があります。

このフィールドを作成ワークフローの後に編集することはできません。

5. (オプション) アプリケーションログパスに、次のように入力します。

- a. アプリケーションログパス

これは、保存するログを含むアプリケーションフォルダまたはファイルへのパス(1つまたは複数)です。各ログパスは、アプリケーションのベースパスを基準にして指定します。この機能を使用すると、各ストリームセッションの最後に、Amazon GameLift Streams は指定されたファイルを、指定された名前の Amazon S3 バケットにコピーします。コピー操作は、アプリケーションフォルダのサブフォルダでは、再帰的に実行されません。

ログ記録を無効にするには、すべてのアプリケーションログパスを削除し、アプリケーションのログ出力先をクリアします。

このフィールドはいつでも編集できます。

- b. アプリケーションログの出力

これは、Amazon GameLift Streams がアプリケーションログファイルをコピーする Amazon S3 バケットの URI です。このフィールドは、アプリケーションログパスを指定する場合に必須です。

ログ記録を無効にするには、すべてのアプリケーションログパスを削除し、アプリケーションのログ出力先をクリアします。

このフィールドはいつでも編集できます。

自動でログファイルを保存するには、Amazon GameLift Streams に S3 バケットへの保存アクセス許可を与える必要があります。ログ記録用のバケットを Amazon GameLift Streams に作成させると、作成時にアクセス許可ポリシーが自動的に適用されます。独自のバケットを指定する場合は、アクセス許可ポリシーを自分で適用する必要があります。

バケットアクセス許可ポリシーテンプレート

次のポリシーコードをコピーして、アプリケーションログに使用するバケットに適用します。必ず amzn-s3-demo-bucket を既存の S3 バケットの名前に置き換えてください。

```
{  
    "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. (オプション) タグで、このアプリケーションにタグを割り当てます。

タグは、 AWS リソースの整理に役立つラベルです。詳細については、[「AWS リソースのタグ付け」](#) を参照してください。

たとえば、アプリケーションバージョンを追跡するには、などのタグを使用します
`application-version : my-game-1121`。

7. [アプリケーションを作成] を選択します。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用してを設定する必要があります AWS リージョン。セットアップ手順については、「」を参照してください[のダウンロード AWS CLI](#)。

を使用してアプリケーションを作成するには AWS CLI

では、コンテンツに合わせてカスタマイズされた [CreateApplication](#) コマンド AWS CLI を使用します。

```
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"
```

この場合、次のようにになります。

- **description:**

人間が読み取れるアプリケーション用のラベルです。この値は一意である必要はありません。ベストプラクティスとして、アプリケーションのわかりやすい説明、名前、またはラベルを使用してください。このフィールドはいつでも編集できます。

- **runtime-environment:**

これは、アプリケーションを実行するランタイム環境です。Amazon GameLift Streams は、Windows、Ubuntu 22.04 LTS、または [Proton](#) のいずれかで実行できます。

このフィールドを作成ワークフローの後に編集することはできません。

次のいずれかのランタイム環境から選択。

- Linux アプリケーションの場合:

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

- Windows アプリケーションの場合:

- Microsoft Windows Server 2022 Base (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:**

これは、Amazon S3 バケット内のアプリケーションのルートフォルダへの Amazon S3 URI です。このフォルダおよびサブフォルダには、ビルド実行ファイルとサポートファイルを含める必要があります。

有効な URI は、アプリケーションの実行とストリーミングに必要なすべてのファイルを含むバケットプレフィックスです。例: というバケットmygamebuildには、ゲームビルドファイルの 3 つの完全なバージョンが含まれており、それぞれが別のフォルダにあります。フォルダ mygamebuild-EN101 内のビルドをストリーミングします。この例では、URI は s3://amzn-s3-demo-bucket/mygamebuild-EN101 です。

このフィールドを作成ワークフローの後に編集することはできません。

- executable-path:

これは、Amazon GameLift Streams がストリーミングする実行可能ファイルの相対パスとファイル名です。への相対パスを指定しますapplication-source-uri。このファイルは、アプリケーションのルートフォルダに含める必要があります。Windows アプリケーションの場合、ファイルは .exe、.cmd、または .bat で終わるファイル名を持つ有効な Windows 実行可能ファイルまたはバッチファイルである必要があります。Linux アプリケーションの場合、ファイルは有効な Linux バイナリ実行可能ファイルであるか、シェバン ('#!') で始まる最初のインターリタ行を含むスクリプトである必要があります。

このフィールドを作成ワークフローの後に編集することはできません。

リクエストが成功すると、Amazon GameLift Streams は次のようなレスポンスを返します。

```
{  
    "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"  
}
```

アプリケーションのステータスを確認するには、次の例に示すように [GetApplication](#) コマンドを呼び出します。

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

Amazon GameLift Streams は、アプリケーションの準備に数分かかります。この間、新しいアプリケーションは処理中ステータスになります。アプリケーションが準備完了ステータスになったら、次のステップ「」に進むことができます [ストリームグループを作成する](#)。

リクエストがエラーを返した場合、またはアプリケーションが作成されてもエラーステータスになった場合は、Amazon S3 と Amazon GameLift Streams の両方へのアクセスを含むユーザー認証情報を使用していることを確認してください。

Note

アプリケーションが準備完了ステータスになると、Amazon GameLift Streams はアプリケーションファイルをプライベート Amazon S3 バケットに正常にコピーしました。新しいアプリケーションに影響を与えることなく、元のアプリケーションファイルを削除できます。これにより、ストレージコストを最適化することもできます。詳細については、「[アプリケーションの削除](#)」を参照してください。

アプリケーションを編集する

Ready ステータスのすべてのアプリケーションの設定を更新できます。既存のアプリケーションに変更を加えると、これらの変更は新規および既存のストリームグループのストリーミング動作に影響します。

Console

Amazon GameLift Streams コンソールでアプリケーションを編集するには

1. ナビゲーションバーで、アプリケーションを選択して既存のアプリケーションのリストを表示します。編集するアプリケーションを選択します。
2. アプリケーションの詳細ページで、変更する設定を含むセクションを見つけ、それに応じてタグの編集または管理を選択します。
3. 以下の設定を指定することができます。

簡単な説明

人間が読み取れるアプリケーション用のラベルです。この値は一意である必要はありません。ベストプラクティスとして、アプリケーションのわかりやすい説明、名前、またはラベルを使用してください。このフィールドはいつでも編集できます。

アプリケーションのログパス

これは、保存するログを含むアプリケーションフォルダまたはファイルへのパス(1つまたは複数)です。各ログパスは、アプリケーションのベースパスを基準にして指定します。この機能を使用すると、各ストリームセッションの最後に、Amazon GameLift Streams は指定されたファイルを、指定された名前の Amazon S3 バケットにコピーします。コピー操作は、アプリケーションフォルダのサブフォルダでは、再帰的に実行されません。

ログ記録を無効にするには、すべてのアプリケーションログパスを削除し、アプリケーションのログ出力先をクリアします。

このフィールドはいつでも編集できます。

アプリケーションのログ出力

これは、Amazon GameLift Streams がアプリケーションログファイルをコピーする Amazon S3 バケットの URI です。このフィールドは、アプリケーションログパスを指定する場合に必須です。

ログ記録を無効にするには、すべてのアプリケーションログパスを削除し、アプリケーションのログ出力先をクリアします。

このフィールドはいつでも編集できます。

自動でログファイルを保存するには、Amazon GameLift Streams に S3 バケットへの保存アクセス許可を与える必要があります。ログ記録用のバケットを Amazon GameLift Streams に作成させると、作成時にアクセス許可ポリシーが自動的に適用されます。独自のバケットを指定する場合は、アクセス許可ポリシーを自分で適用する必要があります。

詳細については、「[アプリケーションログバケットのアクセス許可ポリシー](#)」を参照してください。

[タグ]

タグは、AWS リソースの整理に役立つラベルです。詳細については、[「AWS リソースのタグ付け」](#)を参照してください。

たとえば、アプリケーションバージョンを追跡するには、などのタグを使用します application-version : my-game-1121。

4. [Save changes] (変更の保存) をクリックします。Amazon GameLift Streams コンソールがアプリケーションの詳細ページに戻り、更新された設定が表示されます。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用して を設定する必要があります AWS リージョン。セットアップ手順については、「」を参照してください[のダウンロード AWS CLI](#)。

を使用してアプリケーションを編集するには AWS CLI

では、コンテンツに合わせてカスタマイズされた [UpdateApplication](#) コマンド AWS CLI を使用します。

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

この場合、次のようになります。

- identifier: 編集するアプリケーション。

この値は、アプリケーション[リソースを一意に識別する Amazon リソースネーム \(ARN\)](#) または ID です。

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

ID の例: a-9ZY8X7Wv6

- description:

人間が読み取れるアプリケーション用のラベルです。この値は一意である必要はありません。ベストプラクティスとして、アプリケーションのわかりやすい説明、名前、またはラベルを使用してください。このフィールドはいつでも編集できます。

- **application-log-paths:**

これは、保存するログを含むアプリケーションフォルダまたはファイルへのパス(1つまたは複数)です。各ログパスは、アプリケーションのベースパスを基準にして指定します。この機能を使用すると、各ストリームセッションの最後に、Amazon GameLift Streams は指定されたファイルを、指定された名前の Amazon S3 バケットにコピーします。コピー操作は、アプリケーションフォルダのサブフォルダでは、再帰的に実行されません。

ログ記録を無効にするには、すべてのアプリケーションログパスを削除し、アプリケーションのログ出力先をクリアします。

このフィールドはいつでも編集できます。

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

これは、Amazon GameLift Streams がアプリケーションログファイルをコピーする Amazon S3 バケットの URI です。このフィールドは、アプリケーションログパスを指定する場合に必須です。

ログ記録を無効にするには、すべてのアプリケーションログパスを削除し、アプリケーションのログ出力先をクリアします。

このフィールドはいつでも編集できます。

自動でログファイルを保存するには、Amazon GameLift Streams に S3 バケットへの保存アクセス許可を与える必要があります。ログ記録用のバケットを Amazon GameLift Streams に作成させると、作成時にアクセス許可ポリシーが自動的に適用されます。独自のバケットを指定する場合は、アクセス許可ポリシーを自分で適用する必要があります。

詳細については、「[アプリケーションログバケットのアクセス許可ポリシー](#)」を参照してください。

アプリケーションの削除

不要になったアプリケーションを削除します。このアクションは、Amazon GameLift Streams に保存されているアプリケーションコンテンツファイルを含むアプリケーションを完全に削除します。た

ただし、Amazon S3 バケットにアップロードした元のファイルは削除されません。Amazon GameLift Streams がアプリケーションを作成した後はいつでも削除できます。これは、Amazon GameLift Streams が Amazon S3 バケットにアクセスする唯一の時間です。

削除できるのは、次の条件を満たすアプリケーションのみです。

- アプリケーションは準備完了またはエラー状態です。
- 進行中のストリームセッションでアプリケーションがストリーミングされていない。クライアントがストリームセッションを終了するまで待つか、Amazon GameLift Streams API で [TerminateStreamSession](#) を呼び出してストリームを終了する必要があります。

アプリケーションがストリームグループにリンクされている場合は、削除する前に、関連付けられたすべてのストリームグループからリンクを解除する必要があります。コンソールで、このプロセスを説明するダイアログボックスが表示されます。

Console

Amazon GameLift Streams コンソールを使用してアプリケーションを削除するには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソールを開きます。](#)
2. ナビゲーションバーで、アプリケーションを選択して既存のアプリケーションのリストを表示します。削除するアプリケーションを選択します。
3. アプリケーションの詳細ページで、削除を選択します。
4. 削除ダイアログボックスで、削除アクションを確認します。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用して を設定する必要があります AWS リージョン。セットアップ手順については、「」を参照してください[のダウンロード AWS CLI](#)。

を使用してアプリケーションを削除するには AWS CLI

では、コンテンツに合わせてカスタマイズされた [DeleteApplication](#) コマンド AWS CLI を使用します。

```
aws gameliftstreams delete-application \
```

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

この場合、次のようになります。

- **identifier:** 削除するアプリケーション。

この値は、アプリケーションリソースを一意に識別する Amazon リソースネーム (ARN) または ID です。

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

ID の例: a-9ZY8X7Wv6

Amazon GameLift Streams がアプリケーションの削除を開始します。この間、アプリケーションは Deleting ステータスになります。Amazon GameLift Streams がアプリケーションを削除すると、アプリケーションを取得できなくなります。

アプリケーションログバケットのアクセス許可ポリシー

独自のアプリケーションログ Amazon S3 バケットを指定する場合は、Amazon GameLift Streams がログファイルをバケットに保存できるように、バケットにアクセス許可ポリシーを適用する必要があります。次のテンプレートを使用して、Amazon S3 のアクセス許可を更新します。

バケットアクセス許可ポリシーテンプレート

次のポリシーコードをコピーして、アプリケーションログに使用するバケットに適用します。必ず amzn-s3-demo-bucket を既存の S3 バケットの名前に置き換えてください。

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "PutPolicy",  
            "Effect": "Allow",  
            "Principal": {  
                "Service": [  
                    "gameliftstreams.amazonaws.com"  
                ]  
            },  
            "Action": "s3:PutObject"  
        }  
    ]  
}
```

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

Note

Amazon GameLift Streams は、クロスアカウントリソースアクセスを許可しません。Amazon S3 バケットは、アプリケーションリソースと同じ AWS アカウントによって所有されている必要があります。これはサービスによって強く適用されますが、AWS サービスにアクセス許可を付与するときに [混乱した代理問題](#) を防ぐために、常に aws:SourceAccount または aws:SourceArn 条件を含めることがベストプラクティスです。

リンクされたストリームグループ

同じコンピューティングリソースのプールを使用して複数のアプリケーションをストリーミングする場合は、複数のアプリケーションを同じストリームグループにリンクできます。同様に、異なるコンピューティングリソースのセットを使用してアプリケーションをストリーミングする場合は、アプリケーションを複数のストリームグループにリンクできます。

アプリケーションをストリームグループにリンクする方法の詳細については、「」を参照してください [マルチアプリケーションストリームグループの概要](#)。

Amazon GameLift Streams ストリームグループを使用してストリーミングを管理する

Amazon GameLift Streams アプリケーションをセットアップしたら、コンピューティングリソースを管理およびデプロイして、アプリケーションを実行およびストリーミングする準備が整います。Amazon GameLift Streams ストリームグループは、これらのコンピューティングリソースのコレクションを表します。ストリーム容量をスケーリングすることで、サポートする同時ストリームの最大数を指定します。

Amazon GameLift Streams は、ストリームグループを作成するにコンピューティングリソースを割り当て AWS リージョン ます。リモートロケーションをストリームグループに追加し、ロケーションごとのキャパシティを管理することもできます。ストリームセッションは、地理的にエンドユーザーに近い場所でホストするのがベストプラクティスです。これにより、レイテンシーが最小限に抑えられ、ストリームの品質が向上します。詳細については、[AWS リージョン Amazon GameLift Streams でサポートされている およびリモートロケーション](#) を参照してください。

ストリームグループでは、ストリームグループがストリーミングできる 1 つ以上の Amazon GameLift Streams アプリケーションを指定できます。1 つのアプリケーションを複数のストリームグループに含めることができます。異なる設定またはタイプのコンピューティングリソースを設定して、同じアプリケーションをストリーミングできます。たとえば、アプリケーションをストリーミングするための 2 つのグラフィック品質のオプションを提供するには、異なるストリームクラス設定で 2 つのストリームグループを設定し、それらを同じアプリケーションにリンクできます。

逆に、1 つのストリームグループに複数のアプリケーションを含めることができます。デフォルトアプリケーションは、ストリームグループの作成時に設定でき、追加のリンクされたアプリケーションです。詳細については、[マルチアプリケーションストリームグループの概要](#) を参照してください。

ストリームグループとアプリケーションを関連付ける方法はユースケースによって異なりますが、関係はmany-to-manyになる可能性があります。

重要なサービスの更新と修正を受けるには、3~4 週間ごとにストリームグループを再作成する必要があります。詳細については、[ストリームグループのライフサイクル](#) を参照してください。

トピック

- [ストリーム容量について](#)
- [キャパシティとサービスクォータ](#)
- [ロケーションについて](#)
- [ストリームグループを作成する](#)
- [全般設定の編集](#)
- [容量の編集](#)
- [容量のスケールダウン動作](#)
- [ストリームグループにロケーションを追加する](#)
- [ストリームグループ内の場所を削除する](#)
- [ストリームグループを削除する](#)

- ・[リンクされたアプリケーション](#)
- ・[ストリームグループのライフサイクル](#)
- ・[ストリームグループのメンテナンス](#)

ストリーム容量について

ストリームグループの容量またはストリーム容量を設定することで、エンドユーザーに同時に配信できるストリームの数を管理します。ストリーム容量は、ストリームグループがサポートできる同時ストリームセッションの数を表します。各場所で設定されます。

- ・**常時稼働容量:** ゼロ以外の場合、この設定は、割り当てられた最小ストリーミング容量を示し、サービスに解放されることはありません。使用の有無にかかわらず、常にこの基本レベルの容量に対して料金を支払います。
- ・**最大容量:** これは、サービスが割り当てるができる最大容量を示します。新しく作成されたストリームの開始には数分かかる場合があります。アイドル状態になると、容量はサービスに解放されます。割り当てられた容量は、解放されるまで課金されます。
- ・**ターゲットアイドル容量:** これは、将来のアクティビティを想定してサービスが事前に割り当てて保持するアイドル容量を示します。これにより、容量割り当ての遅延からユーザーを隔離できます。意図的なアイドル状態で保持されている容量に対しては料金が発生します。

ある場所で最大容量が 100 に設定されているストリームグループがある場合、ストリームグループには、その場所で同時に 100 人のエンドユーザーにストリーミングするのに十分なリソースがあることを意味します。ストリーム容量は、ユーザーの需要の変化に合わせて、各場所 (現在のクォータ量まで) でいつでも増減できます。

Amazon GameLift Streams はまず、既に割り当てられているアイドル容量を使用して新しいセッションリクエストを受理しようとします。これにより、アイドル容量がターゲットアイドル容量を下回る場合、新しい容量は非同期的に割り当てられます。使用可能なアイドル容量がない場合、リクエストは一時停止され、ストリームグループの最大容量まで、新しい容量がオンデマンドで割り当てられます。最大容量に達し、使用可能なアイドル容量がない場合、セッションリクエストは既存のセッションが終了し、容量が解放されるのを待ちます。

セッションが終了すると、対応する容量はアイドルとしてマークされます。ターゲットアイドル値よりも多くのアイドル容量がある場合、余分な容量は割り当て解除され、わずかな遅延後にサービスに返されます。アイドル状態の容量が設定された最小値 (ゼロ) を下回る場合、サービスはアイドル状態の容量の割り当てを解除しません。

マルチテナントストリームクラス(コンピューティングリソースごとに 1 つ以上のセッションをストリーミングできます)を持つストリームグループでストリーム容量を指定する場合、容量はテナントの倍数である必要があります。たとえば、gen6n_highストリームクラスのマルチテナントは 2 です。つまり、ストリームグループに割り当てられた各コンピューティングリソースは、2 つのクライアントにストリーミングできます。したがって、リクエストする容量は 2 の倍数である必要があります。

容量のスケーリングは、ストリームグループの合計コストに反映されます。Amazon GameLift Streams のコストを管理するための請求アラートを設定してください。「[請求アラートを作成して、使用状況をモニタリングする](#)」を参照してください。

ストリームグループの容量を変更するには、ストリームグループ設定を編集し、容量設定の新しい値を入力します。常時オン容量を変更すると、Amazon GameLift Streams は、新しいリソースをプロビジョニングするか、既存のリソースをシャットダウンすることで、割り当てられたリソースを新しい値に合わせて調整します。リソースがすぐに利用できない場合、常時オンの容量を増やすには数分以上かかることがあります。常時オンの容量を減らすと、割り当てられたリソースのプロビジョニングが解除されるまでに数分かかります。

例: ストリーム容量設定

次の例は、さまざまなユースケースの一般的なストリーム容量設定を示しています。

1. コスト意識の高い開発フェーズ: コストを削減したい開発者です。Minimum (always-on) capacity = 0、Maximum capacity = 10、Target Idle (pre-warmed) capacity = 1 を設定します。これにより、高速起動に使用できるセッションが少なくとも 1 つ保持されます。
2. 需要が固定された計画イベント: 需要在が既知の計画イベントに対して高速セッションを開始する場合。Minimum (always-on) capacity = 200、Maximum capacity = 200、Target Idle (pre-warmed) capacity = 0 を設定します。200 容量分のみお支払いいただきます。需要がわかっているため、スケーリングの遅延は発生しません。
3. バースト容量を持つ大規模なイベント: ピーク時に 1 分あたり 100 の新しいセッションを持つ 1,000 ユーザーを計画しています。Minimum = 0、Maximum = 1,000、Target Idle = 100 を設定します。これにより、アイドル状態のときにコストを削減できます。これにより、少なくとも 100 セッションを高速起動に使用できます。

Note

`OnDemandCapacity` 入力パラメータは廃止されました。API を使用して容量を設定する場合は、`MaximumCapacity`代わりに を使用します。

キャパシティとサービスクオータ

Amazon GameLift Streams の使用には、アカウントでストリーミングするために設定できる GPUs (コンピューティングリソース) の合計数を制限するサービスクオータが適用されます。デフォルトのクオータとクオータの使用率は、GameLift Streams のサービスクオータコンソールで確認できます。これらのクオータがストリーム容量とどのように相互作用するかを理解することで、ストリーミングインフラストラクチャを計画し、容量の制限を回避できます。

具体的には、GPU サービスクオータは、アカウント内のすべてのストリームグループでロケーションごとにリクエストできる特定のストリームクラスファミリーの GPUs の最大数を指定します。たとえば、アカウントに 5 gen6n GPUs の制限がある場合us-west-2、すべてのストリームグループus-west-2での合計ストリーム容量を提供するために必要な gen6n GPUs の合計は 5 以下である必要があります。これには、常時オン容量とオンデマンド容量の両方の GPUs が含まれます。

これらの GPUs によって提供される合計ストリーム容量を計算する場合、マルチテナントストリームクラスは GPU ごとに複数のセッションのストリーミングをサポートすることに注意してください。したがって、などのストリームグループでマルチテナントストリームクラスを使用している場合はgen6n_high、容量がクオータにどのようにカウントされるかを決定するときに、これを考慮する必要があります。gen6n_ultra やなどのシングルテナントストリームクラスはgen6n_ultra_win2022、ストリームセッションごとに 1 つの GPU を専用にします。

例: クオータが容量に与える影響

次の例は、サービスクオータが複数のストリームグループとロケーションでストリーム容量とやり取りする方法を示しています。この例では、アカウントの場所ごとに 10 gen6n GPUs のクオータがあるとします。

1. シングルテナントストリームグループを作成する: で 5 つの合計容量 (常時オンとオンデマンド) を持つストリームクラスを使用してgen6n_ultraストリームグループを作成しますus-east-2。このストリームクラスには 1:1 のテナンシー (GPU ごとに 1 つのストリーム) があるため、合計容量が 5 GPUs が必要です。これにより、に 5 つの残りの GPUs ますus-east-2。

2. マルチテナントストリームグループを作成する: 合計容量が 6 のストリームクラスを使用して別のgen6n_highストリームグループを作成しますus-east-2。このストリームクラスには 1:2 のテナンシー (GPU あたり 2 つのストリーム) があるため、必要な GPU は 3 GPUs 6 です。これにより、に 2 つの残りの GPUs が残りますus-east-2。
3. 他の場所に容量を追加する: これらのストリームグループを作成した後、に 2 つの残りの GPUs がありますがus-east-2、us-west-2やなどの他の場所では 10 GPUs を使用できますeu-west-1。これらの場所は、前に作成したストリームグループに追加することも、これらの場所を持つ新しいストリームグループを作成することもできます。

この例では、場所ごと、およびすべてのストリームグループにクオータが適用され、サービス制限内に留まりながら複数の地理的リージョンにストリーミング容量を分散できることを示しています。

Note

AWS サービスとして GameLift Streams を選択すると、Service Quotas コンソールで、これらのクオータの使用率を含む Applied アカウントレベルまたはデフォルトのクオータを表示できます。詳細については、「[Amazon GameLift Streams サービスクォータ](#)」を参照してください。

口케ーションについて

の場所は、Amazon GameLift Streams がコンピューティングリソースを割り当ててアプリケーションをホストし、ユーザーにストリーミングする場所です。レイテンシーを短縮し、品質を向上させるには、ユーザーに近い場所を選択する必要があります。デフォルトでは、プライマリ口ケーションと呼ばれるストリームグループを AWS リージョン 作成したからストリーミングできます。さらに、ストリームグループはカバレッジを拡張して、リモート口ケーションと呼ばれるサポートされている他の口ケーションからストリーミングできます。

サポートされている口ケーションの完全なリストについては、「」を参照してください[AWS リージョン およびリモート口ケーション](#)。

マルチ口ケーションストリームグループ

プライマリ口ケーション (ストリームグループを作成した AWS リージョン) に加えて、複数の口ケーションからアプリケーションとストリームセッションをホストするように設定されたストリームグループ。各口ケーションの容量を管理します。

ストリームグループを作成する

Console

Amazon GameLift Streams コンソールでストリームグループを作成するには

- にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソールを開きます](#)。ストリームグループ AWS リージョン を作成する を選択します。このリージョンは、ストリームグループでストリーミングするアプリケーションのリージョンと同じである必要があります。詳細については、「[入門ガイド](#)」の「[リージョンの選択](#)」を参照してください。AWS マネジメントコンソール
- 作成ワークフローを開くには、ナビゲーションペインでストリームグループを選択し、ストリームグループの作成を選択します。
- ストリームグループの定義で、次のように入力します。
 - 説明

人間が読み取れるストリームグループ用のラベル。この値は一意である必要はありません。ベストプラクティスとして、ストリームグループのわかりやすい説明、名前、またはラベルを使用してください。このフィールドはいつでも編集できます。

- タグ

タグは、AWS リソースの整理に役立つラベルです。詳細については、「[AWS リソースのタグ付け](#)」を参照してください。

- Select stream class で、ストリームグループのストリームクラスを選択します。

- ストリームクラスオプション

アプリケーションを実行およびストリーミングするコンピューティングリソースのタイプ。ストリームクラスの選択は、ストリーミングエクスペリエンスの質とコストに影響します。ストリームグループごとに指定できるストリームクラスは 1 つだけです。アプリケーションに最適なクラスを選択します。

ストリームクラス	説明
gen6n_pro _win2022	(NVIDIA、pro) 最も多くのリソースを必要とする 3D シーンの複雑度が非常に高いアプリケーションをサポート。Microsoft Windows Server 2022 Base でアプリケーションを実行

ストリームクラス	説明
	<p>し、DirectX 12 をサポート。Unreal Engine バージョン 5.6 まで、32/64 ビットアプリケーション、チート対策テクノロジーと互換性があります。NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 16 RAM: 64 GB VRAM: 24 GB</p> <p>テナンシー: 最大 1 つの同時ストリームセッションをサポート。</p>
gen6n_pro	<p>(NVIDIA、pro) 最も多くのリソースを必要とする 3D シーンの複雑度が非常に高いアプリケーションをサポート。NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 16 RAM: 64 GB VRAM: 24 GB</p> <p>テナンシー: 最大 1 つの同時ストリームセッションをサポート。</p>
gen6n_ult ra_win2022	<p>(NVIDIA、ultra) 3D シーンの複雑度が高いアプリケーションをサポート。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 をサポート。Unreal Engine バージョン 5.6 まで、32/64 ビットアプリケーション、チート対策テクノロジーと互換性があります。NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 24 GB</p> <p>テナンシー: 最大 1 つの同時ストリームセッションをサポート。</p>

ストリームクラス	説明
gen6n_ultra	<p>(NVIDIA、 ultra) 3D シーンの複雑度が高いアプリケーションをサポート。 NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 24 GB</p> <p>テナンシー: 最大 1 つの同時ストリームセッションをサポート。</p>
gen6n_high	<p>(NVIDIA、 high) 3D シーンの複雑度が中から高のアプリケーションをサポート。 NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 4 RAM: 16 GB VRAM: 12 GB</p> <p>テナンシー: 最大 2 つの同時ストリームセッションをサポート。</p>
gen6n_medium	<p>(NVIDIA、 medium) 3D シーンの複雑度が中程度のアプリケーションをサポート。 NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 2 RAM: 8 GB VRAM: 6 GB</p> <p>テナンシー: 最大 4 つの同時ストリームセッションをサポート。</p>
gen6n_small	<p>(NVIDIA、 small) 3D シーンの複雑度が軽く、 CPU の使用率が低いアプリケーションをサポート。 NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 1 RAM: 4 GB VRAM: 2 GB</p> <p>テナンシー: 最大 12 の同時ストリームセッションをサポート。</p>

ストリームクラス	説明
gen5n_win2022	<p>(NVIDIA、 ultra) 3D シーンの複雑度が非常に高いアプリケーションをサポート。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 と DirectX 11 をサポート。Unreal Engine バージョン 5.6 まで、32/64 ビットアプリケーション、チート対策テクノロジーをサポート。NVIDIA A10G Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 24 GB</p> <p>テナンシー: 1 つの同時ストリームセッションをサポート。</p>
gen5n_high	<p>(NVIDIA、 high) 3D シーンの複雑度が中から高のアプリケーションをサポート。NVIDIA A10G Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 4 RAM: 16 GB VRAM: 12 GB</p> <p>テナンシー: 最大 2 つの同時ストリームセッションをサポート。</p>
gen5n_ultra	<p>(NVIDIA、 ultra) 3D シーンの複雑度が非常に高いアプリケーションをサポート。NVIDIA A10G Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 24 GB</p> <p>テナンシー: 1 つの同時ストリームセッションをサポート。</p>

ストリームクラス	説明
gen4n_win2022	<p>(NVIDIA、 ultra) 3D シーンの複雑度が高いアプリケーションをサポート。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 と DirectX 11 をサポート。Unreal Engine バージョン 5.6 まで、32/64 ビットアプリケーション、チート対策テクノロジーをサポート。NVIDIA T4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 16 GB</p> <p>テナンシー: 1 つの同時ストリームセッションをサポート。</p>
gen4n_high	<p>(NVIDIA、 high) 3D シーンの複雑度が中から高のアプリケーションをサポート。NVIDIA T4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 4 RAM: 16 GB VRAM: 8 GB</p> <p>テナンシー: 最大 2 つの同時ストリームセッションをサポート。</p>
gen4n_ultra	<p>(NVIDIA、 ultra) 3D シーンの複雑度が高いアプリケーションをサポート。NVIDIA T4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 16 GB</p> <p>テナンシー: 1 つの同時ストリームセッションをサポート。</p>

続行するには、次へを選択します。

- Link application で、ストリーミングするアプリケーションを選択するか、「アプリケーションなし」を選択して後で選択します。ストリームグループは、作成後に編集して、アプリケーションを追加または削除できます。Ready ステータスがで、選択したストリームクラスと互換性のあるランタイムを持つアプリケーションのみをリンクできます。デフォルトでは、表に表示されているのは、これらのアプリケーションのみです。Ready ステータスのす

べてのアプリケーションを表示するには、ドロップダウンリストから All runtimes を選択します。

 Note

アプリケーションが表示されない場合は、現在の AWS リージョン 設定を確認してください。アプリケーションをリンクできるのは、同じリージョンにあるストリームグループのみです。

続行するには、次へ を選択します。

6. 「ストリーム設定の構成」の「場所と容量」で、ストリームグループにアプリケーションをストリーミングする容量がある 1 つ以上の場所を選択します。デフォルトでは、プライマリロケーションと呼ばれるストリームグループを作成するリージョンは、既にストリームグループに追加されており、削除できません。追加する各場所の横にあるチェックボックスをオンにして、場所を追加できます。レイテンシーを短縮し、ストリーミングの品質を向上させるには、ユーザーに近い場所を選択する必要があります。

ロケーションごとに、ストリーミング容量を指定できます。[ストリーム容量] は、一度にアクティブにできる同時実行ストリームの数を表します。各ストリームグループの場所ごとにストリーム容量を設定します。

- **常時稼働容量:** ゼロ以外の場合、この設定は、割り当てられた最小ストリーミング容量を示し、サービスに解放されることはありません。使用の有無にかかわらず、常にこの基本レベルの容量に対して料金を支払います。
- **最大容量:** これは、サービスが割り当てることができる最大容量を示します。新しく作成されたストリームの開始には数分かかる場合があります。アイドル状態になると、容量はサービスに解放されます。割り当てられた容量は、解放されるまで課金されます。
- **ターゲットアイドル容量:** これは、将来のアクティビティを想定してサービスが事前に割り当てて保持するアイドル容量を示します。これにより、容量割り当ての遅延からユーザーを隔離できます。意図的なアイドル状態で保持されている容量に対しては料金が発生します。

いずれかの容量を調整することで、場所に対するユーザー需要の変化に合わせて、ストリームの合計容量をいつでも増減できます。Amazon GameLift Streams は、常時オンのキャパシティプールにアイドル状態で事前に割り当てられたリソースがあれば、それを使用してストリーミングリクエストを実行します。すべての常時オン容量が使用されている

場合、Amazon GameLift Streams はオンデマンド容量で指定された最大数まで追加のコンピューティングリソースをプロビジョニングします。割り当てられた容量がスケールするにつれて、その変化がストリームグループの総コストに反映されます。

リンクされたアプリケーションは、有効な各場所に自動的にレプリケートされます。リモート口けーションがストリームをホストする前に、アプリケーションはリモート口けーションでのレプリケーションを終了する必要があります。レプリケーションステータスを確認するには、作成後にストリームグループを開き、リンクされたアプリケーションの表の「レプリケーションステータス」列を参照してください。現在のステータスをクリックすると、追加された各口けーションのレプリケーションステータスが表示されます。

Note

アプリケーションデータは、このストリームグループのプライマリ口けーションを含む、有効なすべての口けーションに保存されます。ストリームセッションデータは、プライマリ口けーションとストリーミングが発生した口けーションの両方に保存されます。

7. 「ストリームグループの確認と作成」で、ストリームグループの設定を確認し、必要に応じて変更を加えます。すべて正しい場合は、ストリームグループの作成を選択します。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用して を設定する必要があります AWS リージョン。セットアップ手順については、「」を参照してください [のダウンロード AWS CLI](#)。

を使用してストリームグループを作成するには AWS CLI

では、コンテンツに合わせてカスタマイズされた [CreateStreamGroup](#) コマンド AWS CLI を使用します。

```
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}]'
```

各一般項の意味は次のとおりです。

description:

人間が読み取れるストリームグループ用のラベル。この値は一意である必要はありません。ベストプラクティスとして、ストリームグループのわかりやすい説明、名前、またはラベルを使用してください。このフィールドはいつでも編集できます。

default-application-identifier

Amazon GameLift [アプリケーションリソースに割り当てられた Amazon リソース名 \(ARN\)](#) の値または ID。アプリケーションは READYステータスである必要があります。

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

ID の例: a-9ZY8X7Wv6

stream-class

ストリームクラスオプション

アプリケーションを実行およびストリーミングするコンピューティングリソースのタイプ。ストリームクラスの選択は、ストリーミングエクスペリエンスの質とコストに影響します。ストリームグループごとに指定できるストリームクラスは 1 つだけです。アプリケーションに最適なクラスを選択します。

ストリームクラス	説明
gen6n_pro _win2022	(NVIDIA、pro) 最も多くのリソースを必要とする 3D シーンの複雑度が非常に高いアプリケーションをサポート。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 をサポート。Unreal Engine バージョン 5.6 まで、32/64 ビットアプリケーション、チート対策テクノロジーと互換性があります。NVIDIA L4 Tensor Core GPU を使用。 アプリケーションあたりのリソース: vCPU: 16 RAM: 64 GB VRAM: 24 GB テナンシー: 最大 1 つの同時ストリームセッションをサポート。

ストリームクラス	説明
gen6n_pro	<p>(NVIDIA、pro) 最も多くリソースを必要とする 3D シーンの複雑度が非常に高いアプリケーションをサポート。NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 16 RAM: 64 GB VRAM: 24 GB</p> <p>テナンシー: 最大 1 つの同時ストリームセッションをサポート。</p>
gen6n_ult ra_win2022	<p>(NVIDIA、ultra) 3D シーンの複雑度が高いアプリケーションをサポート。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 をサポート。Unreal Engine バージョン 5.6 まで、32/64 ビットアプリケーション、チート対策テクノロジーと互換性があります。NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 24 GB</p> <p>テナンシー: 最大 1 つの同時ストリームセッションをサポート。</p>
gen6n_ultra	<p>(NVIDIA、ultra) 3D シーンの複雑度が高いアプリケーションをサポート。NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 24 GB</p> <p>テナンシー: 最大 1 つの同時ストリームセッションをサポート。</p>
gen6n_high	<p>(NVIDIA、high) 3D シーンの複雑度が中から高いアプリケーションをサポート。NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 4 RAM: 16 GB VRAM: 12 GB</p> <p>テナンシー: 最大 2 つの同時ストリームセッションをサポート。</p>

ストリームクラス	説明
gen6n_medium	<p>(NVIDIA、medium) 3D シーンの複雑度が中程度のアプリケーションをサポート。NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 2 RAM: 8 GB VRAM: 6 GB</p> <p>テナンシー: 最大 4 つの同時ストリームセッションをサポート。</p>
gen6n_small1	<p>(NVIDIA、small) 3D シーンの複雑度が軽く、CPU の使用率が低いアプリケーションをサポート。NVIDIA L4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 1 RAM: 4 GB VRAM: 2 GB</p> <p>テナンシー: 最大 12 の同時ストリームセッションをサポート。</p>
gen5n_win2022	<p>(NVIDIA、ultra) 3D シーンの複雑度が非常に高いアプリケーションをサポート。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 と DirectX 11 をサポート。Unreal Engine バージョン 5.6 まで、32/64 ビットアプリケーション、チート対策テクノロジーをサポート。NVIDIA A10G Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 24 GB</p> <p>テナンシー: 1 つの同時ストリームセッションをサポート。</p>
gen5n_ultra	<p>(NVIDIA、ultra) 3D シーンの複雑度が非常に高いアプリケーションをサポート。NVIDIA A10G Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 24 GB</p> <p>テナンシー: 1 つの同時ストリームセッションをサポート。</p>

ストリームクラス	説明
gen5n_high	<p>(NVIDIA、high) 3D シーンの複雑度が中から高のアプリケーションをサポート。NVIDIA A10G Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 4 RAM: 16 GB VRAM: 12 GB</p> <p>テナント: 最大 2 つの同時ストリームセッションをサポート。</p>
gen4n_win2022	<p>(NVIDIA、ultra) 3D シーンの複雑度が高いアプリケーションをサポート。Microsoft Windows Server 2022 Base でアプリケーションを実行し、DirectX 12 と DirectX 11 をサポート。Unreal Engine バージョン 5.6 まで、32/64 ビットアプリケーション、チート対策テクノロジーをサポート。NVIDIA T4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 16 GB</p> <p>テナント: 1 つの同時ストリームセッションをサポート。</p>
gen4n_ultra	<p>(NVIDIA、ultra) 3D シーンの複雑度が高いアプリケーションをサポート。NVIDIA T4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 8 RAM: 32 GB VRAM: 16 GB</p> <p>テナント: 1 つの同時ストリームセッションをサポート。</p>
gen4n_high	<p>(NVIDIA、high) 3D シーンの複雑度が中から高のアプリケーションをサポート。NVIDIA T4 Tensor Core GPU を使用。</p> <p>アプリケーションあたりのリソース: vCPU: 4 RAM: 16 GB VRAM: 8 GB</p> <p>テナント: 最大 2 つの同時ストリームセッションをサポート。</p>

location-configurations

このストリームグループに追加する一連の場所とその容量。デフォルトでは、容量が指定されていない場合、Amazon GameLift Streams は、ストリームグループが作成された場所で 1 つのストリームを開始するのに十分な常時オンストリーム容量のみを割り当てます。Amazon GameLift Streams がサポートするロケーションの完全なリストについては、「」を参照してください[AWS リージョン Amazon GameLift Streams でサポートされている およびリモート ロケーション](#)。

容量の値は、ストリームグループのストリームクラスのテナンシー値の整数倍である必要があります。

リクエストが成功すると、Amazon GameLift Streams は次のようなレスポンスを返します。

```
{  
    "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 は、未割り当てのコンピューティングリソースを検索し、新しいストリームグループのプロビジョニングを開始します。これには数分かかる場合があります。この間、新しいストリームグループは Activating ステータスになります。

ステータスがアクティブの場合、ストリームグループの容量を調整できます。詳細については、[容量 の編集](#) を参照してください。

ストリームグループがアクティブステータスになると、ストリーミング用のリソースをデプロイする準備が整います。ストリーミングを開始するには、「」を参照してください[Amazon GameLift Streams でストリームセッションを開始する](#)。

全般設定の編集

Amazon GameLift Streams は、ストリームグループ設定のコンソールで、ステータス、ストリームグループ ID、説明、ストリームグループ ARN、ストリームクラスをグループ化します。これらのうち、新しいストリームグループを作成せずに更新できるのは説明のみです。

Console

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソールを開きます。](#)
2. ナビゲーションバーで、ストリームグループを選択して既存のストリームグループのリストを表示します。編集するストリームグループを選択します。
3. ストリームグループの詳細ページで、設定の編集を選択します。
4. 説明を更新するには、新しい値を入力します。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用して を設定する必要があります AWS リージョン。セットアップ手順については、「」を参照してください[のダウンロード AWS CLI。](#)

を使用してストリームグループの説明を編集するには AWS CLI

では、コンテンツに合わせてカスタマイズされた [UpdateStreamGroup](#) コマンド AWS CLI を使用します。

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

各一般項の意味は次のとおりです。

identifier

ストリームグループ[リソースを一意に識別する Amazon リソースネーム \(ARN\)](#) または ID。

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

ID の例: sg-1AB2C3De4

description

人間が読み取れるストリームグループ用のラベル。この値は一意である必要はありません。ベストプラクティスとして、ストリームグループのわかりやすい説明、名前、またはラベルを使用してください。このフィールドはいつでも編集できます。

容量の編集

各ロケーションの容量を調整して、ストリームグループをスケールします。

ストリームグループ容量クォータ [Amazon GameLift Streams サービスクォータ](#) の詳細については AWS アカウント、「」を参照してください。

Console

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソールを開きます。](#)
2. ナビゲーションバーで、ストリームグループを選択して既存のストリームグループのリストを表示します。編集するストリームグループを選択します。
3. ストリームグループの詳細ページで、設定の編集を選択します。
4. 場所ごとに、テーブル内の関連するセルに新しい常時オン容量、最大容量、ターゲットアイドル容量の値を入力します。容量の値は、ストリームグループのストリームクラスのテナシーカー値の整数倍である必要があります。

常時オンの容量値をゼロに設定すると、ストリームグループはストリームにホストを割り当てません。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用して を設定する必要があります AWS リージョン。セットアップ手順については、「」を参照してください [のダウンロード AWS CLI](#)。

を使用してストリーム容量を編集するには AWS CLI

では、コンテンツに合わせてカスタマイズされた [UpdateStreamGroup](#) コマンド AWS CLI を使用します。

```
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}]'
```

各一般項の意味は次のとおりです。

identifier

ストリームグループリソースを一意に識別する Amazon リソースネーム (ARN) または ID。

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

ID の例: `sg-1AB2C3De4`

location-configurations

このストリームグループで新しい容量で更新する一連の場所。容量の値は、ストリームグループのストリームクラスのテナンシー値の整数倍である必要があります。

ストリームグループのロケーションの容量を更新すると、Amazon GameLift Streams はリクエストの処理を開始します。これには時間がかかる場合があります。この間、Amazon GameLift Streams は、設定した必要な常時オンストリーム容量を満たすために、必要に応じてストリームグループ内のリソースを割り当てまたは解放します。ストリーム容量のプロビジョニングステータスを表示するには、Amazon GameLift Streams コンソールでストリームグループの詳細ページを表示するか、[GetStreamGroup API](#) を呼び出します。

ストリームグループがアクティブステータスで、使用可能なストリーム容量があり、アプリケーションがストリーミングする場所へのレプリケーションを完了したら、ストリーミングを開始できます。詳細については、[Amazon GameLift Streams でストリームセッションを開始する](#) を参照してください。

容量のスケールダウン動作

容量をスケールダウンすると、Amazon GameLift Streams はホストがアイドル状態になるまで待機してから解放します。ホストは 1 つまたは 2 つのセッションをサポートできるため、ホスト上のすべてのアクティブなセッションが終了した場合にのみ、ホストはアイドル状態になります。ストリー

ムセッションは、ユーザーがセッションを終了するか、セッションがタイムアウトすると終了します。したがって、既存のセッションが最大許容期間に達すると、希望する容量に達するまでに最大24時間かかる場合があります。ストリームグループ内のすべてのアクティブなストリームセッションを強制終了する場合は、コンソールで、または [DeleteStreamGroup API](#) を使用してストリームグループを削除するか、[TerminateStreamSession API](#) を使用してアクティブなセッションを一度に1つずつ終了できます。

ストリームグループにロケーションを追加する

Console

Amazon GameLift Streams コンソールを使用してストリームグループにロケーションを追加するには

1. ナビゲーションバーで、ストリームグループを選択して既存のストリームグループのリストを表示します。新しい場所を追加するストリームグループを選択します。
2. ストリームグループの詳細ページで、設定の編集を選択します。
3. このストリームグループに追加する場所の横にあるチェックボックスをオンにし、容量を設定します(複数可)。
4. ストリーム容量のコストなど、選択したロケーションの概要を確認します。保存を選択して選択を確定します。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用してを設定する必要があります AWS リージョン。セットアップ手順については、「」を参照してください[のダウンロード AWS CLI](#)。

を使用してストリームグループにロケーションを追加するには AWS CLI

では、コンテンツに合わせてカスタマイズされた [AddStreamGroupLocations](#) コマンド AWS CLI を使用します。

```
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}]'
```

各一般項の意味は次のとおりです。

identifier

ストリームグループリソースを一意に識別する Amazon リソースネーム (ARN) または ID。

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

ID の例: sg-1AB2C3De4

location-configurations

このストリームグループに追加する一連の場所とその容量。Amazon GameLift Streams がサポートするロケーションの完全なリストについては、「」を参照してください[AWS リージョン Amazon GameLift Streams でサポートされている およびリモートロケーション](#)。

容量の値は、ストリームグループのストリームクラスのテナンシー値の整数倍である必要があります。

アプリケーションが新しいロケーション（複数可）へのレプリケーションを完了し、ストリームグループに使用可能なストリーム容量がある場合、新しいロケーション（複数可）からストリーミングを開始できます。ストリーミングの詳細については、「」を参照してください[Amazon GameLift Streams でストリームセッションを開始する](#)。Amazon GameLift Streams はリクエストの処理を開始します。この間、Amazon GameLift Streams はアプリケーションをレプリケートし、コンピューティングリソースを新しい場所に割り当てます。レプリケーションのステータス列のステータスをクリックすると、ストリームグループの詳細ページのリンクされたアプリケーションセクションからレプリケーションのステータスを表示できます。

ストリームグループ内の場所を削除する

特定の場所からのコンピューティングリソースの使用を停止するには、ストリームグループからその場所を削除できます。ストリームグループのプライマリロケーションを削除することはできません。ただし、その場所にコンピューティングリソースが必要ない場合は、ストリーム容量をゼロに設定できます。

⚠ Warning

ストリームグループ内の場所を削除すると、Amazon GameLift Streams はその場所のアクティブなストリームを切断し、接続されたエンドユーザーのストリームを停止します。

Console

Amazon GameLift Streams コンソールを使用してストリームグループからロケーションを削除するには

1. ナビゲーションペインで、ストリームグループを選択して既存のストリームグループのリストを表示します。
2. ロケーションを削除するストリームグループの名前を選択します。
3. ストリームグループの詳細ページで、設定の編集を選択します。
4. 削除する場所の名前の横にあるチェックボックスをオフにします。
5. [保存] を選択します。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用して を設定する必要があります AWS リージョン。セットアップ手順については、「」を参照してください[のダウンロード AWS CLI](#)。

を使用してストリームグループから場所を削除するには AWS CLI

で、コンテンツに合わせてカスタマイズされた [RemoveStreamGroupLocations](#) コマンド AWS CLI を使用します。

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

各一般項の意味は次のとおりです。

identifier

ストリームグループ[リソースを一意に識別する Amazon リソースネーム \(ARN\)](#) または ID。

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

ID の例: sg-1AB2C3De4

locations

このストリームグループから削除する一連の場所。Amazon GameLift Streams がサポートするロケーションの完全なリストについては、「」を参照してください[AWS リージョン Amazon GameLift Streams でサポートされている およびリモートロケーション。](#)

ストリームグループを削除する

任意のステータスのストリームグループを削除できます。このアクションは、ストリームグループを完全に削除し、そのコンピューティングリソースを解放します。進行中のストリームがある場合、このアクションはストリームを停止し、エンドユーザーはストリームを表示できなくなります。

ベストプラクティスとして、ストリームグループを削除する前に、進行中のストリームをチェックし、それらを停止するステップを実行します。

Console

Amazon GameLift Streams コンソールを使用してストリームグループを削除するには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソールを開きます。](#)
2. 既存のストリームグループのリストを表示するには、ナビゲーションペインでストリームグループを選択します。
3. 削除するストリームグループの名前を選択します。
4. ストリームグループの詳細ページで、削除を選択します。
5. 削除ダイアログボックスで、削除アクションを確認します。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用して を設定する必要があります AWS リージョン。セットアップ手順については、「」を参照してください[のダウンロード AWS CLI。](#)

を使用してストリームグループを削除するには AWS CLI

では、コンテンツに合わせてカスタマイズされた [DeleteStreamGroup](#) コマンド AWS CLI を使用します。

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

各一般項の意味は次のとおりです。

identifier

ストリームグループ [リソースを一意に識別する Amazon リソースネーム \(ARN\)](#) または ID。

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

ID の例: sg-1AB2C3De4

Amazon GameLift Streams は、コンピューティングリソースのリリースとストリームグループの削除を開始します。この間、ストリームグループは削除ステータスになります。Amazon GameLift Streams がストリームグループを削除すると、ストリームグループを取得できなくなります。

リンクされたアプリケーション

同じコンピューティングリソースのプールを使用して複数のアプリケーションをストリーミングする場合は、複数のアプリケーションを同じストリームグループにリンクできます。同様に、コンピューティングリソースの異なるセットを使用してアプリケーションをストリーミングする場合は、アプリケーションを複数のストリームグループにリンクできます。

アプリケーションをストリームグループにリンクする方法の詳細については、「」を参照してください[マルチアプリケーションストリームグループの概要](#)。

ストリームグループのライフサイクル

ストリームグループの最大有効期間は 365 日です。ベストプラクティスとして、重要なサービスの更新と修正を受け取り、最適なパフォーマンスを確保するために、3~4 週間ごとにストリームグループを再作成することをお勧めします。ストリームグループを再作成しても、アップロードされたアプリケーションには影響しません。

ストリームグループの経過時間に応じて、次の制限が適用されます。

- 180 日後: 新しいアプリケーションの関連付けでストリームグループを更新できなくなります

- 365 日: ストリームグループの有効期限が切れ、セッションをストリーミングできなくなります

ストリームグループに関連付けられたアカウントは、から 2 つのリマインダー通知を受け取ります AWS Health。1 つは 45 日間、もう 1 つは 150 日間です。これらの通知は、アプリケーションの関連付け機能が 180 日に失われることを知らせるものです。また、335 日に 1 つの最終通知があり、ストリームグループが 365 日に期限切れになることを知らせます。メンテナンス警告は、AWS Health ダッシュボードと Amazon GameLift Streams コンソールのストリームグループページにも表示されます。

ストリームグループの有効期限を確認するには、コンソールでストリームグループの詳細ページを表示するか、[GetStreamGroup API レスポンス](#)のExpiresAtフィールドを使用します。

期限切れのストリームグループのステータスは EXPIREDで、読み取り専用になります。更新したり、新しいストリームセッションを開始したりすることはできません。機能を回復するには、ストリームグループを再作成します。

ストリームグループのメンテナンス

新しいストリームグループの使用を必要とする機能がリリースされるたびに、ストリームグループの詳細ページの上部に「メンテナンスが必要」というメッセージが表示され、古いことが通知されます。ストリームグループの再作成は手動プロセスですが、これを行うには、メッセージのストリームグループの作成ボタンを使用してプロセスを開始します。一部のフィールドは自動的に入力されます。

ストリームグループが 180 日以上経過している場合も、ストリームグループのメンテナンスが必要です。再作成されるまで、新しいアプリケーションをこれらの古いストリームグループにリンクすることはできなくなります。365 日後、ストリームグループからのストリーミングは実行できず、ストリームグループへの変更は許可されません。

マルチアプリケーションストリームグループの概要

マルチアプリケーションストリームグループは、複数のアプリケーションにリンクされたストリームグループです。これにより、単一のストリームグループで同じコンピューティングリソースのセットを使用して、複数のアプリケーションをストリーミングできます。

マルチアプリケーションストリームグループの一般的なユースケースは、ゲームのさまざまなバージョンをリリースすることです。たとえば、ストリームグループを作成し、デフォルトのアプリケーションをゲームの元のバージョンに設定するとします。次に、ゲームの他のバージョンを含む追加の

アプリケーションを作成し、ストリームグループにリンクするとします。これらのアプリケーションは同じストリームグループに関連付けられているため、これらのゲームをすべてストリーミングするには、1セットのコンピューティングリソースまたはストリーム容量を管理するだけで済みます。つまり、エンドユーザーストリームのアプリケーションに関係なく、アプリケーションはこのストリームグループが割り当てたのと同じセットのコンピューティングリソースで実行されます。

その他の実際の例を以下に示します。

- ・ さまざまなストリーミング層を顧客に提供するゲームストリーミングプラットフォーム。
- ・ ゲームの複数のバージョンをテストしている品質保証チーム。
- ・ 複数のアプリケーションに単一のストリームグループを使用することで、ストリーム容量管理を簡素化します。
- ・ 一連のアプリケーションが同じストリーム容量のプールからストリーミングできるようにするには。

制限事項と要件

アプリケーションを関連付けることができるのは、互換性のあるランタイム環境とストリームクラスを持つストリームグループのみです。詳細については、[ランタイム環境とストリームクラスの互換性](#)を参照してください。

アプリケーションとストリームグループには、次の関連付け制限が適用されます。これらの制限は、すべてのお客様のサービス内で修正されます。

名前	デフォルト	引き上げ可能	説明
ストリームグループ内のアプリケーション	250	不可	ストリームグループに関連付けることができる Amazon GameLift Streams アプリケーションの最大数。
アプリケーションあたりのストリームグループの関連付け	100	不可	Amazon GameLift Streams アプリケーションに関連付けることができるストリームグループの最大数。

デフォルトアプリケーションについて

各ストリームグループにはデフォルトアプリケーションが 1 つあり、これが最初にストリームグループに追加されるアプリケーションです。デフォルトアプリケーションは、常時稼働しているすべてのコンピューティングリソースに自動的に事前にキャッシュされるため、ストリームの起動時のアプリケーションのロード時間を短縮できます。Amazon GameLift Streams サービスは、最適化プロセス中に他のリンクされたアプリケーションもキャッシュできます。

デフォルトアプリケーションおよびその他のリンクされたアプリケーションの特徴:

- ・ デフォルトのアプリケーションは、ストリームの起動中のアプリケーションのロード時間を短縮するために、事前にキャッシュされています（常時オンの容量など、事前に割り当てられたコンピューティングリソース上）。
- ・ デフォルトのアプリケーションは変更できます。ストリームグループでデフォルトアプリケーションを切り替える場合、新しいデフォルトアプリケーションがすべての場所で事前キャッシュされるまでに最大数時間かかることがあります。
- ・ ストリームグループからストリーミングを開始する前に、少なくとも 1 つのリンクされたアプリケーションが必要です。リンクされた最初のアプリケーションは自動的にデフォルトのアプリケーションになります。
- ・ ストリームグループのデフォルトアプリケーションのリンクを解除すると、Amazon GameLift Streams は、関連付けられた残りのアプリケーションがあれば、自動的に新しいデフォルトアプリケーションを選択します。
- ・ 同じアプリケーションを複数のストリームグループのデフォルトアプリケーションにすることができます。
- ・ リンクされたアプリケーションのセットは、ストリームグループが 180 日経過するまで変更可能です。実際には、これは、ストリームグループが 180 日経過するまでアプリケーションをリンクおよびリンク解除できることを意味します。その後、ストリームグループの残りのライフサイクルを通じてのみ、ストリームグループからアプリケーションのリンクを解除できます。

デフォルトのアプリケーションを変更する

最初のアプリケーションをストリームグループにリンクすると、自動的にデフォルトのアプリケーションになり、事前キャッシュの利点が得られます。デフォルトのアプリケーションはいつでも変更でき、これらの利点を別のアプリケーションに与えることができます。

Note

ストリームグループでデフォルトアプリケーションを切り替える場合、新しいデフォルトアプリケーションがすべての場所で事前キャッシュされるまでに最大数時間かかることがあります。

Console

Amazon GameLift Streams コンソールを使用してデフォルトのアプリケーションを変更するには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソールを開きます。](#)
2. ナビゲーションバーで、ストリームグループを選択して既存のストリームグループのリストを表示します。
3. ストリームグループを選択して詳細を表示します。
4. リンクされたアプリケーションで、デフォルトにするアプリケーションを選択します。
5. デフォルトにする を選択します。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用して を設定する必要があります AWS リージョン。セットアップ手順については、「」を参照してください[のダウンロード AWS CLI](#)。

を使用してデフォルトのアプリケーションを変更するには AWS CLI

では、コンテンツに合わせてカスタマイズされた [UpdateStreamGroup](#) コマンド AWS CLI を使用します。デフォルトにするアプリケーションは、既にストリームグループに関連付けられている必要があります。

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

各一般項の意味は次のとおりです。

- **identifier:**

デフォルトにするアプリケーションがあるストリームグループ。

この値は、ストリームグループ[リソースを一意に識別する Amazon リソースネーム \(ARN\)](#) または ID することができます。

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

ID の例: sg-1AB2C3De4

- **default-application-identifier:**

このストリームグループでデフォルトにするアプリケーション。

この値は、アプリケーション[リソースを一意に識別する Amazon リソースネーム \(ARN\)](#) または ID です。

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

ID の例: a-9ZY8X7Wv6

アプリケーションをリンクする

アプリケーションをストリームグループにリンクまたは関連付けると、ストリームグループはアプリケーションをストリーミングできます。ストリームグループの最初のアプリケーションである場合、自動的にデフォルトのアプリケーションになります。180 日が経過するまで、追加のアプリケーションをストリームグループにリンクおよびリンク解除できます。その後、グループの残りのライフサイクルを通じてのみ、ストリームグループからアプリケーションのリンクを解除できます。

A Important

180 日以上経過したストリームグループにアプリケーションをリンクすることはできません。異なるアプリケーションをストリームグループに関連付けるには、まずストリームグループを再作成する必要があります。ストリームグループを再作成する方法については、「」を参照してください[ストリームグループのメンテナンス](#)。

アプリケーションをリンクする前に、ストリームグループがアクティブステータスであることを確認します。

Console

Amazon GameLift Streams コンソールを使用してリンクするには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソールを開きます。](#)
2. ナビゲーションバーで、ストリームグループを選択して既存のストリームグループのリストを表示します。
3. ストリームグループを選択して詳細を表示します。
4. リンクされたアプリケーションで、リンクアプリケーションを選択します。
5. リンクするアプリケーションを選択します。選択内容を確認し、アプリケーションのリンクを選択します。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用して を設定する必要があります AWS リージョン。セットアップ手順については、「」を参照してください[のダウンロード AWS CLI。](#)

を使用してアプリケーションをリンクするには (複数可) AWS CLI

では、コンテンツに合わせてカスタマイズされた [AssociateApplications](#) コマンド AWS CLI を使用します。

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

各一般項の意味は次のとおりです。

- **identifier:**

これらのアプリケーションをリンクするストリームグループ。

この値は、ストリームグループ [リソースを一意に識別する Amazon リソースネーム \(ARN\)](#) または ID にすることができます。

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

ID の例: sg-1AB2C3De4

- **application-identifiers:**

このストリームグループにリンクする一連のアプリケーション。

この値は、アプリケーション [リソースを一意に識別する Amazon リソースネーム \(ARN\)](#) または ID です。

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

ID の例: a-9ZY8X7Wv6

アプリケーションのリンクを解除する

ストリームグループからアプリケーションのリンクを解除または関連付けを解除すると、そのストリームグループに割り当てられたコンピューティングリソースを使用してこのアプリケーションをストリーミングできなくなります。処理中のストリームは終了するまで継続されるため、エンドユーザーのストリームの中断を回避できます。Amazon GameLift Streams は、このストリームグループを使用して新しいストリームを開始しません。リンク解除アクションは、ストリームグループのストリーム容量には影響しません。

ストリームグループのデフォルトアプリケーションのリンクを解除すると、Amazon GameLift Streams は、関連付けられた残りのアプリケーションがあれば、自動的に新しいデフォルトアプリケーションを選択します。

Console

Amazon GameLift Streams コンソールを使用してリンクを解除するには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソールを開きます。](#)

2. ナビゲーションバーで、ストリームグループを選択して既存のストリームグループのリストを表示します。
3. ストリームグループを選択して詳細を表示します。
4. リンクされたアプリケーションで、リンクを解除するアプリケーション(複数可)を選択します。アプリケーションのリンク解除を選択します。
5. Unlink applications ダイアログで、Unlink アクションを確認し、Unlink を選択します。

CLI

前提条件

ユーザー認証情報と選択した AWS CLI を使用して AWS リージョンを設定する必要があります。セットアップ手順については、「」を参照してください。[AWS CLI のダウンロード](#)

を使用してアプリケーションのリンクを解除するには(複数可) AWS CLI

では、コンテンツに合わせてカスタマイズされた [DisassociateApplications](#) コマンド AWS CLI を使用します。

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

各一般項の意味は次のとおりです。

- **identifier:**

これらのアプリケーションのリンクを解除するストリームグループ。

この値は、ストリームグループ [リソースを一意に識別する Amazon リソースネーム \(ARN\)](#) または ID にすることができます。

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

ID の例: sg-1AB2C3De4

- **application-identifiers:**

このストリームグループからリンクを解除する一連のアプリケーション。

この値は、アプリケーションリソースを一意に識別する Amazon リソースネーム (ARN) または ID です。

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

ID の例: a-9ZY8X7Wv6

Amazon GameLift Streams でストリームセッションを開始する

このセクションでは、ストリームセッション、エンドユーザーまたはプレイヤーがアプリケーションとやり取りしたりゲームをプレイしたりできるストリームの実際のインスタンスについて説明します。独自のストリームセッションをテストし、ストリームセッションのライフサイクルを理解する方法について説明します。

エンドユーザーにストリームセッションを起動するには、Amazon GameLift Streams を独自のサービスに統合する必要があります。詳細については、「[Amazon GameLift Streams バックエンドサービスとウェブクライアント](#)」を参照してください。

ストリームセッションについて

ストリームセッションを開始するための前提条件は、準備完了ステータスのアプリケーション、ストリーミングする場所で使用可能な容量を持つストリームグループ、およびストリーミングする場所にレプリケートされたアプリケーションです。ストリームセッションは、ストリームグループが割り当てたコンピューティングリソースのいずれかで実行されます。ストリームを開始するときは、ARN または ID 値を使用してストリーミングするストリームグループとアプリケーションを指定する必要があります。

ストリームセッションを正常に開始すると、そのストリームセッションの一意の識別子を受け取ります。次に、その ID を使用してストリームセッションをエンドユーザーに接続します。詳細については、Amazon GameLift Streams API リファレンスの[StartStreamSession](#)」を参照してください。

GameLift

コンソールでのストリームのテスト

Amazon GameLift Streams コンソールを使用してアプリケーションストリームがどのようにであるかをテストする最も直接的な方法。ストリームを開始すると、Amazon GameLift Streams はスト

リームグループが割り当てるコンピューティングリソースのいずれかを使用します。そのため、ストリームグループで使用可能な容量が必要です。

Amazon GameLift Streams コンソールでストリームをテストするには

1. にサインイン AWS マネジメントコンソールし、[Amazon GameLift Streams コンソール](#)を開きます。
2. ストリームは、いくつかの方法でテストできます。ストリームグループページまたはテストストリームページから開始し、以下の手順に従います。
 - a. ストリーミングに使用するストリームグループを選択します。
 - b. ストリームグループページから開始する場合は、ストリームのテストを選択します。テストストリームページから開始する場合は、選択を選択します。これにより、選択したストリームグループのストリーム設定のテストページが開きます。
 - c. リンクされたアプリケーションで、アプリケーションを選択します。
 - d. Location で、使用可能な容量がある場所を選択します。
 - e. (オプション) プログラム設定で、起動時にアプリケーションに渡すコマンドライン引数または環境変数を入力します。
 - f. 選択内容を確認し、ストリームのテストを選択します。
3. ストリームがロードされたら、ストリームで次のアクションを実行できます。
 - a. マウス、キーボード、ゲームパッド (テストストリームではサポートされていないマイクを除く) などの入力を接続するには、入力をアタッチを選択します。カーソルをストリーム ウィンドウに移動すると、マウスが自動的にアタッチされます。
 - b. ストリーミングセッション中に作成されたファイルをセッションの終了時に Amazon S3 バケットにエクスポートするには、ファイルのエクスポートを選択し、バケットの詳細を指定します。エクスポートされたファイルは、セッションページにあります。
 - c. ストリームを全画面で表示するには、全画面表示を選択します。Escape を押して、このアクションを元に戻します。
4. ストリームを終了するには、セッションの終了を選択します。ストリームが切断されると、ストリーム容量が別のストリームを開始できるようになります。

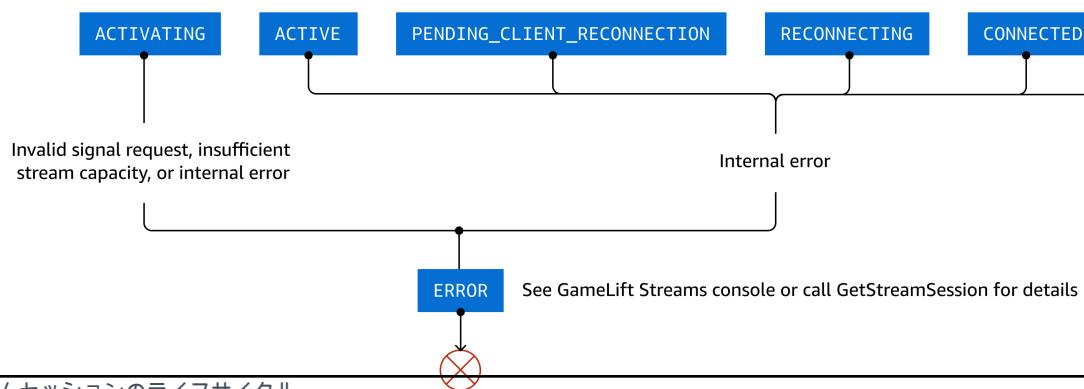
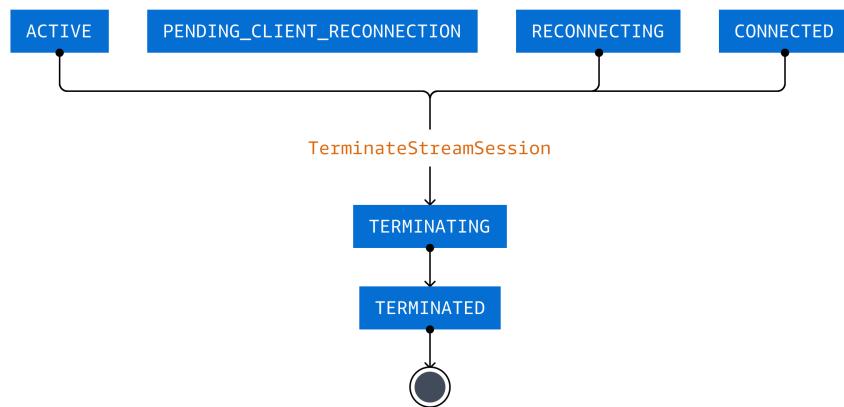
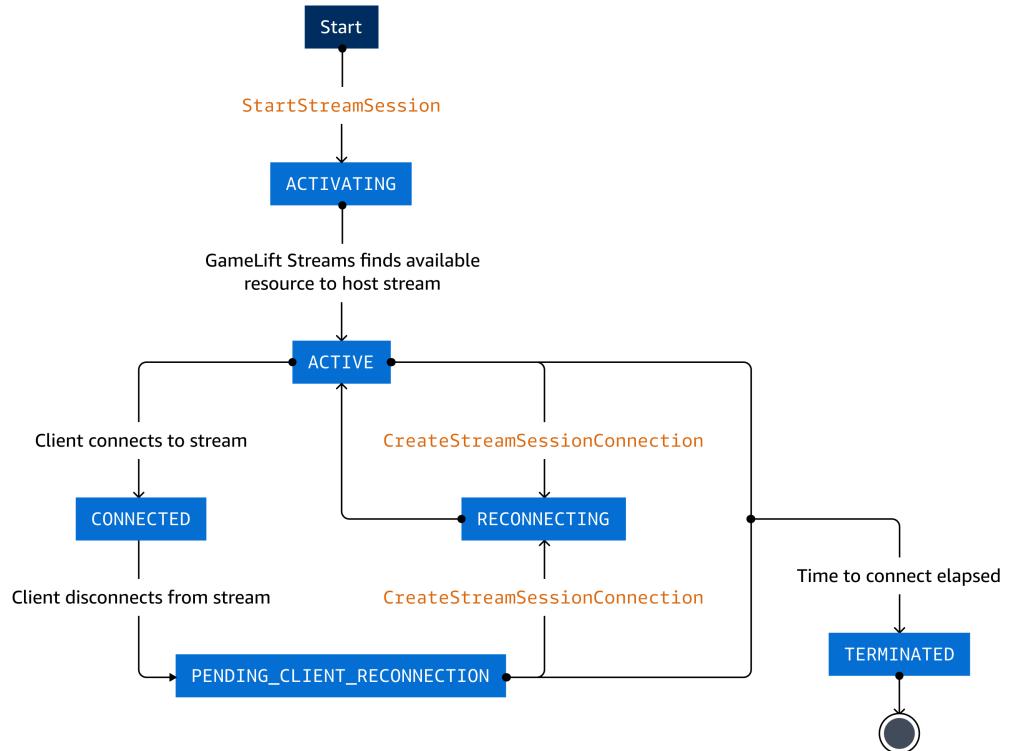
Note

Amazon GameLift Streams コンソールのテストストリーム機能はマイクをサポートしていません。

ストリームセッションのライフサイクル

Amazon GameLift Streams でストリームセッションを操作する場合、この図は、ストリームセッションがライフサイクル全体で移行するさまざまな状態を理解するのに役立ちます。

- [StartStreamSession](#) は、 ACTIVATING 状態から始まる新しいストリームセッションを作成します。Amazon GameLift Streams がストリームをホストするために利用可能なリソースを見つけると、ストリームセッションは に移行します ACTIVE。クライアントがアクティブなストリームに接続すると、ストリームセッションは に移行します CONNECTED。
- クライアントがストリームから切断すると、ストリームセッションは PENDING_CLIENT_RECONNECTION 状態に移行します。[CreateStreamSessionConnection](#) はストリームセッションを に移行し RECONNECTING、クライアントがストリームに再接続するか、新しいストリームセッションを作成します。ストリームセッションがクライアントに再接続する準備ができたら、 に移行します ACTIVE。クライアントが再接続すると、 に戻ります CONNECTED。クライアントがより長く切断された場合 ConnectionTimeoutSeconds、ストリームセッションは終了します。
- クライアントが で指定された期間内に ACTIVE または PENDING_CLIENT_RECONNECTION 状態のストリームセッションに接続しない場合 ConnectionTimeoutSeconds、 に移行します TERMINATED。
- [TerminateStreamSession](#) はストリームの終了を開始し、ストリームセッションは TERMINATING 状態に移行します。ストリームセッションが正常に終了すると、 に移行します TERMINATED。
- を除く任意の状態のストリームセッションは TERMINATED、 に移行できます ERROR。API コールが Status 値 ERROR として を返す場合は、 StatusReason の値でエラーの原因の簡単な説明を確認します。[GetStreamSession](#) を呼び出して、これらの値を確認することもできます。



ストリームセッションに影響するタイムアウト値

ストリームセッションは、セッションライフサイクルのさまざまな側面を制御するいくつかのタイムアウト値によって管理されます。ストリームセッションのライフサイクル中に通常発生する可能性がある のほぼ時系列の順序で、以下が含まれます。

配置タイムアウト

Amazon GameLift Streams が利用可能な容量を使用してストリームセッションをホストするコンピューティングリソースを見つけるための制限時間。配置タイムアウトは、ストリームリクエストを満たすために使用される容量タイプによって異なります。

- 常時オン容量: 75 秒
- オンデマンドキャパシティ:
 - Linux/Proton ランタイム: 90 秒
 - Windows ランタイム: 10 分
- 動作: Amazon GameLift Streams がこの時間内に利用可能なリソースを識別できない場合、ストリームセッションは StatusReason の ERROR で Status が変わります placementTimeout。

接続タイムアウト

Amazon GameLift Streams がクライアントがストリームセッションに接続または再接続するまで待機する時間の長さ。

- パラメータ: ConnectionTimeoutSeconds [StartStreamSession](#)
- 範囲: 1 ~ 3600 秒 (1 時間)
- デフォルト: 120 秒 (2 分)
- 動作: ストリームセッションが ACTIVE または PENDING_CLIENT_RECONNECTION ステータスに達するとタイマーが開始されます。タイムアウト前に接続するクライアントがない場合、セッションは Status が TERMINATED になります。

アイドルタイムアウト

ユーザー入力なしでストリームセッションをアイドル状態にできる時間の長さ。

- パラメータ: (内部)
- 値: 60 分
- 動作: この間にユーザー入力が受信されない場合、Amazon GameLift Streams はストリームセッションを終了します。

セッションの長さのタイムアウト

Amazon GameLift Streams の最大期間は、ストリームセッションを開いたままにします。

- パラメータ: SessionLengthSeconds [StartStreamSession](#)
- 範囲: 1 ~ 86400 秒 (24 時間)
- デフォルト: 43200 秒 (12 時間)
- 動作: 制限時間に達したときに、既存のクライアント接続に関係なくストリームセッションを終了します。

ストリームセッションの終了

ストリームセッションを強制終了する必要がある場合は、次のオプションがあります。

- TerminateStreamSession API を使用する : [TerminateStreamSession](#) を使用するには、ストリームグループ ID とストリームセッション ID が必要です。[ListStreamSessions](#) または [ListStreamSessionsByAccount](#) を --status CONNECTED パラメータとともに使用して、クライアントが接続されているストリームセッションのリストを取得できます。
- ストリームグループからセッションの場所を削除する : セッションがストリーミングされているストリームグループから場所を削除すると、その場所内のすべてのアクティブなストリームセッションが終了します。ストリームグループ内の場所は、コンソールから削除するか、[RemoveStreamGroupLocations](#) API を使用して削除できます。
- セッションのストリームグループを削除する : ストリームグループを削除すると、ストリームグループのすべての場所ですべてのアクティブなストリームセッションが終了します。ストリームグループは、コンソールから、または [DeleteStreamGroup](#) API を使用して削除できます。クライアント接続は突然終了するため、注意が必要です。

ストリームセッションへの再接続

クライアントがセッションを終了せずにストリームセッションから切断された場合、ストリームセッションが開始され ConnectionTimeoutSeconds たときに指定された時間内にセッションに再接続できます。セッションに再接続するには、ストリームセッションの ID が必要です。詳細については、Amazon GameLift Streams API リファレンスの「[CreateStreamSessionConnection](#)」を参照してください。GameLift [React Starter Sample](#) でストリームセッションに再接続する例を確認できます。

ストリームセッションファイルのエクスポート

ストリームセッション中、アプリケーションは、ログ、診断情報、クラッシュダンプ、ファイルの保存、ユーザーデータ、スクリーンショットなど、アプリケーションのデバッグまたは検証に役立つ出力ファイルを生成できます。ストリームセッションファイルのエクスポート機能は、セッション中に作成または変更されたファイルを収集し、圧縮された ZIP ファイルとして指定された Amazon S3 の場所にエクスポートします。また、この機能は、エクスポート ZIP ファイルに含まれているセッションのパフォーマンス統計を 1 秒ごとに収集します。

Warning

ファイルをエクスポートする前に、次の点に注意してください。

- ファイルには、認証情報など、アプリケーションによって書き込まれた機密情報が含まれている場合があります。
- ファイルサイズはアプリケーションサイズによっては大きくなる場合があり、Amazon S3 ストレージコストに影響します。
- ストリームグループのリージョン AWS リージョンとは異なるで Amazon S3 バケットを選択すると、エクスポートされたストリームセッションファイルはリージョン間で移動します。

仕組み

アクティブなストリームセッションでこのオペレーションを手動で呼び出して、そのセッション中に生成されたファイルをエクスポートする必要があります。ストリームセッションは、特に、ACTIVE、CONNECTED、PENDING_CLIENT_RECONNECTION のいずれかのステータスでアクティブである必要があります RECONNECTING。セッションの終了時に、Amazon GameLift Streams はファイルを Amazon Simple Storage Service (Amazon S3) のバケットにエクスポートします。したがって、エクスポートされたすべてのデータはお客様の所有権内であり、Amazon S3 バケットのアクセス許可ポリシーの対象となります。

エクスポートファイルを有効にしたストリームセッションライフサイクルのチュートリアルを次に示します。

1. Amazon GameLift Streams は、コンピューティングリソースで実行されているアプリケーションにユーザーを接続することでセッションを開始します。

2. アプリケーションはストリーミング中に、ランタイム環境のファイルシステムにファイルを作成または変更します。
3. セッションが終了すると、Amazon GameLift Streams はファイルシステム内のすべての新規または変更されたファイルのコピーを取得し、そのファイルを Amazon S3 バケットにエクスポートします。

Amazon GameLift Streams は、以下の生成および変更されたファイルを収集します。.zip アーカイブ内の対応するフォルダでそれらを見つけます。

- application/: アプリケーションまたはゲームが保存されているフォルダ。
- profile/: ユーザーのプロファイルフォルダには、ユーザーの個人設定、設定、データが含まれます。
- temp/: システムの一時フォルダには、アプリケーションとシステムが作成する一時ファイルとデータが含まれています。これには、キャッシュファイル、ログファイル、または中間処理データが含まれます。
- stats/: perf_stats_v1.csv 1 秒あたりに収集されたセッションのパフォーマンス統計を保持するが含まれます。これには、アプリケーションレベルの統計 (CPU とメモリの使用率) とシステムレベルの統計 (CPU、メモリ、GPU、VRAM の使用率) が含まれます。CSV ファイルに含まれる各統計の詳細な説明については、「」を参照してください。[the section called “パフォーマンス統計リファレンス”](#)

ファイルを削除するには、Amazon S3 バケット内の オブジェクトを削除します。

コストへの影響

Amazon S3 にファイルを保存すると、コストが発生します。ストリームセッションは、アプリケーションによっては大量のデータを生成する場合があります。この機能が有効になっている多くのストリームセッションでは、コストが加算される可能性があることに注意してください。

詳細については、[Amazon S3の料金](#)」を参照してください。

ファイルのエクスポート (コンソール)

Amazon GameLift Streams コンソールでストリームセッションファイルのエクスポートを有効にするには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソール](#)を開きます。
2. ナビゲーションバーで、セッションを選択して、過去 90 日間のアクティブなストリームセッションと最近のストリームセッションのリストを表示します。
3. アクティブセッションタブで、アクティブなストリームセッションを選択します。
4. ファイルのエクスポートを選択して、そのストリームセッションのファイルのエクスポート機能を有効にします。
5. ストリームセッションファイルのエクスポートダイアログボックスで、新しい S3 バケットを作成する または既存の S3 バケットを選択する を選択します。コンソールの手順に従って、エクスポートしたデータを保存する S3 オブジェクトを作成または選択します。

⚠ Warning

ZIP ファイル名がディレクトリ内の既存のファイル名と一致する場合、前のファイル名は上書きされます。

6. [確認] を選択します。これで、エクスポートされたファイルタブにセッションが一覧表示されます。
7. セッションが終了し、ファイルがエクスポートされるまで待ちます。

Amazon GameLift Streams は、セッションが終了状態になるとファイルをエクスポートします。セッションが終了すると、アクティブセッションタブから最近のセッションタブに移動します。

エクスポートプロセスのステータスは、セッションエクスポートタブで確認できます。ステータスが保留中の場合、ストリームセッションはまだアクティブであるため、Amazon GameLift Streams はまだファイルをエクスポートしていません。ステータスが成功の場合、提供されたリンクを使用して Amazon S3 からファイルをダウンロードできます。ステータスが Failed の場合、ステータスにカーソルを合わせると、失敗の理由が表示されます。

ファイルのエクスポート (CLI)

前提条件

ユーザー認証情報と選択した AWS CLI を使用して を設定する必要があります AWS リージョン。

セットアップ手順については、「」を参照してください[のダウンロード AWS CLI](#)。

でストリームセッションファイルをエクスポートするには AWS CLI

では、コンテンツに合わせてカスタマイズされた [ExportStreamSessionFiles](#) コマンド AWS CLI を使用します。

```
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
```

各パラメータの意味は次のとおりです。

identifier

ストリームグループ[リソースを一意に識別する Amazon リソースネーム \(ARN\)](#) または ID。

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

ID の例: sg-1AB2C3De4

stream-session-identifier

ストリームセッション[リソースを一意に識別する Amazon リソースネーム \(ARN\)](#) または ID。

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

ID の例: ABC123def4567

output-uri

Amazon GameLift Streams がこのストリームセッションの圧縮エクスポートされたファイルのセットをアップロードする Amazon S3 バケット URI。

指定できる有効な形式は 2 つあります。URI に .zip または .ZIP ファイル拡張子がある場合、Amazon GameLift Streams はエクスポートされたファイルを指定された URI に保存します。それ以外の場合、Amazon GameLift Streams は圧縮フォルダの名前を生成し、URI に保存し

ます。生成された名前はパターンに従いますdate-time-applicationId-streamGroupId-streamSessionId。例えば、次のようにになります。

- という URI を指定すると s3://amzn-s3-demo-bucket/MyGame_Session1.zip、Amazon GameLift Streams はその正確な ZIP フォルダにファイルを保存します。
- という URI を指定すると s3://amzn-s3-demo-bucket/MyGame_Session1/、Amazon GameLift Streams はファイルを に保存します s3://amzn-s3-demo-bucket/ MyGame_Session1/YYYYMMDD-HHMMSS-applicationId-streamGroupId-sessionId.zip。

ZIP ファイル名が、Amazon Simple Storage Service ユーザーガイドの[オブジェクトキーの命名ガイドライン](#)に準拠していることを確認してください。

 Warning

ZIP ファイル名がディレクトリ内の既存のファイル名と一致する場合、前のファイル名は上書きされます。

[GetStreamSession](#) API を呼び出すことで、アクティブなセッションのステータスを確認できます。ストリームセッションの概要から、エクスポートされたファイルのステータスに関する詳細を取得できます。ステータスが保留中の場合、ストリームセッションはまだアクティブであるため、Amazon GameLift Streams はまだファイルをエクスポートしていません。ステータスが成功の場合は、出力 URI に移動して Amazon S3 のファイルを確認します。ステータスが Failed の場合は、StatusReason のを確認します [ExportFilesMetaData](#)。

Amazon GameLift Streams バックエンドサービスとウェブクライアント

Amazon GameLift Streams を使用すると、ウェブブラウザを介してアプリケーションをストリーミングできます。Amazon GameLift Streams Web SDK を使用すると、バックエンドストリーミングサービスをセットアップできます。次に、エンドユーザーはウェブクライアントを介してストリームに接続します。クラウドを通じてゲームをプレイしたり、アプリケーションとやり取りしたりできます。

Amazon GameLift Streams Web SDK には、バックエンドサービスの作成を開始するために使用できるサンプルバックエンドサーバーとサンプルウェブクライアントが含まれています。これらのサンプルを使用して、追加の開発なしで Amazon GameLift Streams のストリーミング方法をテストすることもできます。開始するには、「」を参照してください [Amazon GameLift Streams を使用したウェブサーバーとクライアントのセットアップ](#)。

トピック

- [サポートされているブラウザと入力](#)
- [必要なポート](#)
- [Amazon GameLift Streams を使用したウェブサーバーとクライアントのセットアップ](#)
- [ストリームの外観をカスタマイズする](#)
- [口ケール設定](#)
- [マウスの移動処理](#)
- [アプリケーションとウェブクライアント間のデータチャネル通信](#)

サポートされているブラウザと入力

Amazon GameLift Streams ストリームとその互換性のある入力周辺機器を表示するためにサポートされているプラットフォームとブラウザを以下に示します。ブラウザは、H.264 とも呼ばれるアドバンストビデオコーディング (AVC) とも互換性がある必要があります。

全体として、特にゲームコントローラーとの最高のエンドユーザーエクスペリエンスと最大の互換性を実現するために、Google Chrome、Microsoft Edge、またはカスタム Chromium ベースのデスクトップアプリケーションをお勧めします。

どのコントローラーがどのブラウザと互換性があるかの詳細については、[「Web Gamepad API」](#)を参照してください。Amazon GameLift Streams には一部のガイダンスが適用されない場合がありますが、ほとんどのゲームコントローラーは Bluetooth 経由で正常に接続することが予想されます。

オペレーティングシステム	ブラウザ	Input
Server	Chrome、Edge	キーボード、マウス、マイク、ゲームコントローラー(触覚フィードバックを含む)
	Firefox	キーボード、マウス、マイク、ゲームコントローラー
Mac	Chrome、Edge、Safari	キーボード、マウス、マイク、ゲームコントローラー(Bluetooth モード)(触覚フィードバックを含む)
	Firefox	キーボード、マウス、マイク
Linux	Chrome、Edge、Firefox	キーボード、マウス
Android	Chrome、Edge	シンプルなtouch-to-mouseエミュレーション、マイク、外部物理マウス、キーボード、ゲームコントローラー(Bluetooth モード)

オペレーティングシステム	ブラウザ	Input
iOS	Chrome、Edge、Firefox、Safari	シンプルなtouch-to-mouse工ミニューション、マイク、外部物理マウス、キーボード、ゲームコントローラー(Bluetoothモード)

既知の問題

ブラウザと入力に関する既知の問題は次のとおりです。

- Safari は、 Esc が押されるたびにすぐに全画面表示を終了します。これは上書きできません。
- LinkedIn、Yelp、Instagram などのモバイルアプリ内などの「埋め込み」または「アプリ内」ブラウザビューは、iOS ではサポートされていません。これらは、リアルタイムのインタラクティブストリーミングに必要なブラウザ WebRTC サポートを無効にする傾向があります。標準以外のブラウザ文字列を検出し、Safari でを開くようにユーザーにお勧めします。
- アプリケーションの画面解像度が 1080p に設定されていない場合、マウスの追跡に影響する可能性があります。可能であれば、他の解像度の選択を無効にすることをお勧めします。また、ウインドウモードを無効にし、全画面でのみ実行することをお勧めします。
- Proton でゲームコントローラーのプラグアンドプレイをサポートするには、ネイティブ Linux アプリケーションでサポートされていないにもかかわらず、Proton ランタイム環境で実行されているゲームには、クライアントに接続されても、接続されているゲームコントローラーが常に表示されます。これは、コントローラーがアイドル状態で使用されていない場合でも、コントローラーの入力を求めるゲームの問題である可能性があります。ゲームでは、最後の入力方法に基づいて入力 UI を表示することをお勧めします。

制限事項

- ほとんどのランタイム環境は、Ubuntu 22.04 LTS を除くゲームコントローラーをサポートしています。ゲームコントローラーのサポートが必要な場合は、別のランタイム環境を使用してゲームを

作成することを検討してください。他のランタイム環境のリストについては、「」を参照してください [ランタイム環境](#)。

- PlayStation 5 および Luna ゲームコントローラーは Firefox ではサポートされていません。
- 触覚フィードバックのサポート:
 - PlayStation 4 および Xbox Series S/X コントローラーに関する触覚フィードバックは、Chrome、Edge、Safari でサポートされています。
 - PlayStation 5 DualSense コントローラーの触覚は、Safari ブラウザでのみサポートされています。
 - Firefox は、コントローラーに関する触覚フィードバックをサポートしていません。
 - Android および iOS デバイスは、コントローラーに関する触覚フィードバックをサポートしていません。
- Amazon GameLift Streams コンソールのテストストリーム機能はマイクをサポートしていません。

IPv6 サポート

IPv6-only クライアントへのストリーミングは、Windows ランタイムアプリケーションでのみサポートされています。

実行時間	IPv4 経由のストリーミング	IPv6 経由のストリーミング
Microsoft Windows Server 2022 Base	はい	はい
Ubuntu 22.04 LTS	はい	なし
Proton ランタイム	はい	なし

必要なポート

Amazon GameLift Streams を統合するには、ネットワークインフラストラクチャに必要なポートが開いていてアクセス可能であることを確認します。以下は、Amazon GameLift Streams と通信するためにネットワークで開く予定のポートのリストです。

ポート	プロトコル	目的
443	(HTTPS) TCP	Amazon GameLift Streams を含む AWS APIs
33435-33465	UDP	ウェブ RTC

Amazon GameLift Streams を使用したウェブサーバーとクライアントのセットアップ

このチュートリアルでは、Amazon GameLift Streams のストリーミングサービスを統合するウェブクライアントアプリケーションを設定します。次に、Amazon GameLift Streams Web SDK、JavaScript ライブラリ、および開始できるサンプルコードを使用します。サンプルコードには、シンプルな Amazon GameLift Streams バックエンドウェブサーバーとシンプルなウェブクライアントが含まれています。このチュートリアルを終了すると、サンプルコードを使用してストリームを開始できます。

Amazon GameLift Streams を初めて使用する場合は、[Amazon GameLift Streams で最初のストリームを開始する](#)Amazon S3 にゲームをアップロードし、ブラウザの Amazon GameLift Streams コンソール内からストリーミングをテストするチュートリアルから始めることをお勧めします。

前提条件

- プログラムによるアクセスに適切な認証情報を持つ AWS アカウント。詳細については、「[開発者としての Amazon GameLift Streams のセットアップ](#)」を参照してください。
- AWS SDK。
- Amazon GameLift Streams がサポートするウェブブラウザ — 「」を参照してください[サポートされているブラウザと入力](#)。
- Node.js — [Node.js ダウンロード](#)ページを参照してください。

Web SDK のダウンロード

このチュートリアルでは、[入門製品ページ](#)のリソースセクションから次のマテリアルをダウンロードする必要があります。

- Amazon GameLift Streams Web SDK バンドル: これには、シンプルなバックエンドサービスとウェブクライアントのサンプルコードが含まれます。

- Amazon GameLift Streams Web SDK API リファレンス: この API リファレンスは、JavaScript 用の Amazon GameLift Streams API ラッパーを文書化します。

ストリーミングリソースをセットアップする

ストリームを開始するには、アプリケーションとストリームグループというストリームリソースが必要です。具体的には、以下が必要です。

- 準備完了ステータスのアプリケーション。
- 使用可能なストリーム容量を持つアクティブステータスのストリームグループ。
- プライマリロケーション以外のロケーションでのストリーミングの場合、アプリケーションはそのロケーションへのレプリケーションを完了している必要があります。

Amazon GameLift Streams コンソールまたは Amazon GameLift Streams CLI を使用してアプリケーションとストリームグループを設定するには、[Amazon GameLift Streams ストリームグループを使用してストリーミングを管理する](#) それぞれ [Amazon GameLift Streams でアプリケーションを準備すると](#) を参照してください。または、Amazon GameLift Streams コンソールでの end-to-end のチュートリアルについては、「」を参照してください[Amazon GameLift Streams で最初のストリームを開始する。](#)

バックエンドサーバーをセットアップする

バックエンドサーバーは、ユーザーの認証、ストリームパラメータの設定、エンドユーザーに代わって Amazon GameLift Streams サービス API コールの実行などのタスクを処理します。セットアップの詳細については、サンプルコードと Amazon GameLift Streams Web SDK API リファレンスを参照してください。具体的には、Amazon GameLift Streams Web SDK パッケージの server.js ファイルを参照してください。

Important

このコードはテストおよび評価のみを目的としたサンプルコードであり、本番稼働用容量では使用しないでください。

サンプルバックエンドサービスを実行するには

1. ターミナルまたはコマンドプロンプトを開き、フォルダに移動します。AmazonGameLiftStreamsWebSDK\GameLiftStreamsSampleGamePublisherService\。
2. 以下のコマンドを実行します。

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

サンプルバックエンドサービスを実行すると、エンドユーザーはウェブクライアントを介してストリームに接続できます。次のステップでウェブクライアントをテストします。

ウェブクライアントを起動する

ウェブクライアントアプリケーションは、Amazon GameLift Streamsストリームの受信とデコード、エンドユーザーへのストリーミング、エンドユーザーがアプリケーションとやり取りするためのウェブブラウザUIの提供を担当します。JavaScript Amazon GameLift Streams Web SDKを独自のウェブクライアントアプリケーションに統合する方法の詳細については、サンプルコードとAmazon GameLift Streams Web SDK APIリファレンスを参照してください。具体的には、Amazon GameLift Streams Web SDKパッケージpublic/index.htmlの「」を参照してください。ブラウザでウェブクライアントを起動するときに、ウェブページのソースを確認することもできます。

Note

Amazon GameLift StreamsのWindowsランタイムは、IPv4またはIPv6経由のストリームセッションをサポートします。ただし、LinuxおよびProtonランタイム環境はIPv4経由のストリーミングのみをサポートします。

ウェブクライアントアプリケーションを起動するには

1. ウェブブラウザを開き、に移動します`http://localhost:port/`。ポート番号はバックエンドサーバーによって設定されます。デフォルトでは、これはHTTPポート8000です。
2. ゲームをプレイするか、ソフトウェアを使用します。

- a. マウスなどの入力をアタッチするには、入力をアタッチを選択します。
- b. ゲームを終了するには、Esc キーを選択します。
- c. サーバープロセスを停止するには、Ctrl+C キーを選択します。

ストリーミングリソースをクリーンアップする

Warning

ストリームグループがストリーミング容量を割り当てた場合、その容量が未使用であってもコストが発生します。不要なコストを回避するには、ストリームグループを必要なサイズにスケールします。開発時には、ストリームグループの常時オン容量とターゲットアイドル容量を、使用していないときはゼロにスケーリングすることをお勧めします。詳細については、[ストリームグループを容量ゼロにスケールする](#) を参照してください。

チュートリアルを完了し、アプリケーションをストリーミングする必要がなくなったら、以下の手順に従って Amazon GameLift Streams リソースをクリーンアップします。

ストリームグループの削除

ストリームグループを削除すると、Amazon GameLift Streams はすべてのストリーム容量を解放するように動作します。

Amazon GameLift Streams コンソールを使用してストリームグループを削除するには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソール](#)を開きます。
2. 既存のストリームグループのリストを表示するには、ナビゲーションペインでストリームグループを選択します。
3. 削除するストリームグループの名前を選択します。
4. ストリームグループの詳細ページで、削除を選択します。
5. 削除ダイアログボックスで、削除アクションを確認します。

Amazon GameLift Streams は、コンピューティングリソースのリリースとストリームグループの削除を開始します。この間、ストリームグループは削除ステータスになります。Amazon GameLift Streams がストリームグループを削除すると、そのグループを取得できなくなります。

アプリケーションの削除

削除できるのは、以下の条件を満たすアプリケーションだけです。

- ・ [準備完了] 状態または [エラー] 状態のアプリケーション。
- ・ 進行中のどのストリームセッションでもストリーミングされていないアプリケーション。クラウドアントがストリームセッションを終了するまで待つか、Amazon GameLift Streams API の [TerminateStreamSession](#) を呼び出してストリームを終了する必要があります。

アプリケーションがいずれかのストリームグループにリンクされている場合は、削除する前に、関連するすべてのストリームグループからリンクを解除する必要があります。コンソールでは、このプロセスを説明するダイアログボックスが表示されます。

Amazon GameLift Streams コンソールを使用してアプリケーションを削除するには

1. にサインイン AWS マネジメントコンソール し、[Amazon GameLift Streams コンソール](#)を開きます。
2. ナビゲーションバーで、アプリケーションを選択して既存のアプリケーションのリストを表示します。削除するアプリケーションを選択します。
3. アプリケーションの詳細ページで、削除を選択します。
4. 削除ダイアログボックスで、削除アクションを確認します。

Amazon GameLift Streams がアプリケーションの削除を開始します。この間、アプリケーションは Deletingステータスになります。Amazon GameLift Streams がアプリケーションを削除すると、アプリケーションを取得できなくなります。

ストリームの外観をカスタマイズする

画面のロード

顧客がウェブブラウザを開いてストリームを表示すると、ウェブクライアントは Amazon GameLift Streams ストリームセッションへの接続の確立を開始します。ストリームセッションのロード中に、顧客の画面にカスタム背景とロゴを表示できます。

Amazon GameLift Streams Web SDK サンプルクライアントは、[GameLiftStreamsSampleGamePublisherService/public>LoadingScreen/loadingscreen.js](#) ファイルのフロントエンドウェブクライアントにアニメーションロゴを実装す

る方法を示しています。デフォルトのロード画面は、背景と前景の 2 つの画像で構成されます。前景画像は中央に配置され、パルスアニメーションがあります。アニメーションは、ストリームセッションの接続中にのみ再生されます。

ロード画面を有効にするには

1. Amazon GameLift Streams Web SDK サンプルクライアントで、GameLiftStreamsSampleGamePublisherService/public>LoadingScreen/フォルダに移動します。
2. デフォルト名 および を使用して、背景イメージBackground.pngとフォアグラウンドイメージを追加しますLoadingLogo.png。名前を変更する場合、または別のイメージ形式を使用する場合は、でコードを更新する必要がありますGameLiftStreamsSampleGamePublisherService/public/loadingscreen.js。
3. (オプション) でGameLiftStreamsSampleGamePublisherService/public/loadingscreen.js、JavaScript コードを更新してさまざまなアニメーションを実装します。

口ケール設定

Amazon GameLift Streams では、ストリームごとに口ケール設定を設定できます。これは、アプリケーションが時間や通貨などの場所固有の情報をエンドユーザーのオペレーティングシステムから取得する場合に便利です。

Amazon GameLift Streams は、次の言語をサポートしています。

値	説明
en_US	米国英語 (デフォルト)
ja_jp.UTF-8	Japanese

口ケール設定を変更するには

Amazon GameLift Streams API を使用して [StartStreamSession](#) を呼び出す場合は、`LANG=<language>` を に追加しますAdditionalEnvironmentVariables。口ケール設定はユーザーごとに一意であるため、ストリームセッションレベルで設定します。これを設定しない場合、ストリームはデフォルトで米国英語を使用します。

Example 例

```
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"}'
```

マウスの移動処理

マウスの移動処理は、ストリーミングされたアプリケーションで応答性と直感的なユーザーエクスペリエンスを提供するために不可欠です。Amazon GameLift Streams は、アプリケーションのカーソル動作に基づいてマウス入力送信を自動的に最適化し、カーソルが非表示でも表示でもマウスの動きが自然に感じられるようにします。Amazon GameLift Streams がマウスイベントを処理する方法を理解すると、ストリーミングサービスとシームレスに連携し、可能な限り最高のユーザーエクスペリエンスを提供するアプリケーションを設計できます。

マウス入力モード

Amazon GameLift Streams は、マウスイベントをアプリケーションに送信するために 2 つの異なるモードを使用し、カーソルの可視性に基づいて適切なモードを自動的に選択します。

相対モード

相対モードでは、マウスの更新は前の位置との小さな増分差として送信されます。このモードは、ファーストパーソンシューティング (FPS) ゲームや 3D 方向を使用するインターフェイスなど、正確で継続的なマウスの動き追跡を必要とするアプリケーションに最適です。Amazon GameLift Streams は、オペレーティングシステムのカーソルが非表示または完全に透過的である場合、相対モードを使用します。

絶対モード

絶対モードでは、マウスカーソルの位置は正確な画面座標として送信されます。このモードは、point-and-click ゲームやクリック可能な要素を持つ UI など、正確なカーソル配置に依存するアプリケーションに適しています。Amazon GameLift Streams は、アプリケーションにカスタムカーソルイメージが表示されていても、オペレーティングシステムのカーソルが表示されるときに絶対モードを使用します。

この自動選択により、手動設定を必要とせずに、さまざまなアプリケーションタイプに最適なパフォーマンスが保証されます。

ポインタロック

ポインタロックは、特定の要素内のマウスカーソルをキャプチャし、カーソルを非表示にして、指定された領域からマウスカーソルを離れないようにするウェブ API 機能です。この機能は、視認可能なカーソルの気が散ったり、ウィンドウエッジに到達したりすることなく、カメラ制御や狙いを定めるために無制限のマウスの動きを必要とするゲームに特に役立ちます。

Amazon GameLift Streams は、Web SDK の `InputConfiguration` インターフェイスの `autoPointerLock` プロパティを通じて自動ポインタロック機能を提供します。この機能は [requestPointerLock API](#) と統合され、直感的でコンテキスト対応のマウスキャプチャを提供します。

自動ポインタロックの動作

Amazon GameLift Streams は、アプリケーションが全画面表示で、リモートカーソルがストリームホストに表示されない場合に、ポインタロックを自動的に有効にします。この動作は、一般的なゲーム開発パターンとよく一致します。

- FPS/TPS ゲームと 3D 方向制御 - ポインタは自動的にロックされ、カーソルは非表示になり、FPS ゲームプレイに不可欠な無制限のカメラ制御を提供します。
- Point-and-click ゲームと UI コントロール - ゲームがメニューインタラクションまたは戦略ゲームプレイにカーソルを表示すると、ポインタは表示およびロック解除されたままになり、意図したユーザーエクスペリエンスが保持されます。

設定オプション

`autoPointerLock` プロパティは、次の値を受け入れます。

`true`

リモートカーソルが表示されない場合、マウスは常にキャプチャされます。

`false`

カーソルの可視性に関係なく、マウスはキャプチャされません。

`'fullscreen'` (デフォルト)

マウスは、ビデオ要素が全画面表示モードにあり、リモートカーソルが表示されない場合にのみキャプチャされます。

⚠️ Important

`autoPointerLock` プラットフォームの制限により、`PointerLockEvent` は Safari ブラウザや iOS プラットフォームには影響しません。

ベストプラクティス

ストリーミングされたアプリケーションでマウスを最適に処理するには：

- 常に全画面ストリーミングする - サービスで正常に動作するには、アプリケーションがすでに全画面モードで実行されている必要があります。さらに、ブラウザのサポートを使用して、ストリームをフルスクリーン要素にして、最適なエンドユーザー エクスペリエンスを実現することをお勧めします。これにより、システムカーソルとソフトウェアカーソル間のアライメントの問題などの問題を回避できます。
- 相対モーションのカーソルを非表示にする - アプリケーションでマウスの相対モーション (FPS 形式のカメラコントロールやドラッグベースのインタラクションなど) が予想される場合は、それらのインタラクション中にオペレーティングシステムのカーソルを非表示にします。シナリオによつては、マウスダウンでカーソルを非表示にし、マウスアップで再度表示する必要がある場合があります。
- 絶対配置のカーソルを表示する - アプリケーションで UI インタラクションに正確なカーソル配置が必要な場合は、オペレーティングシステムのカーソルが表示されたままにして絶対座標モードを有効にします。
- さまざまな入力シナリオをテストする - Amazon GameLift Streams がカーソルの可視性の変化に基づいてモードを切り替える可能性があるため、アプリケーションが相対モードと絶対モードの両方を正しく処理していることを確認します。
- さまざまなウィンドウモードをテストする - 該当する場合は、ウィンドウモードと全画面モードの両方でアプリケーションのマウス処理をテストします。入力 `autoPointerLock` 設定に最適な設定を決定します。

アプリケーションとウェブクライアント間のデータチャネル通信

データチャネルを使用すると、Amazon GameLift Streams アプリケーションとウェブクライアント (エンドユーザーのウェブブラウザで実行されている JavaScript コード) の間で任意のメッセージを安全に通信できます。これにより、エンドユーザーはストリームを表示しているウェブブラウザを

介して、Amazon GameLift Streams ガストリーミングしているアプリケーションとやり取りできます。

Amazon GameLift Streams のデータチャネルのユースケースの例を次に示します。

- ・ ユーザーはローカルブラウザでアプリケーションで URLs を開くことができます。
- ・ ユーザーはクリップボード内のコンテンツをアプリケーションに前後に渡すことができます。
- ・ ユーザーはローカルマシンからアプリケーションにコンテンツをアップロードできます。
- ・ 開発者は、アプリケーションにコマンドを送信する UI をブラウザに実装できます。
- ・ ユーザーはスキーマを渡して、視覚化レイヤーの表示を制御できます。

特徴

メッセージのサイズ制限

Amazon GameLift Streams Web SDK は、メッセージあたり 64 KB (65536 バイト) の最大サイズ制限を課します。これにより、メッセージサイズ制限がほとんどのブラウザと互換性があり、通信がストリームの合計帯域幅に与える影響が低くなります。

メトリクス

データチャネル使用状況のメトリクスは、ストリームセッションが終了すると AWS アカウントに送信されます。詳細については、「Amazon GameLift ストリームのモニタリング」セクション [データチャネル](#) の「」を参照してください。 GameLift

データチャネルの使用

Amazon GameLift Streams Web SDK は、アプリケーションにバイト配列としてメッセージを送信する `sendApplicationMessage` 関数を提供します。メッセージは、定義したコールバック関数によって処理 `clientConnection.applicationMessage` されます。

アプリケーションがデータチャネルポートに接続する前にクライアントがメッセージを送信すると、メッセージはキューに入れられます。次に、アプリケーションが接続すると、メッセージを受信します。ただし、クライアントがデータチャネルポートに接続する前にアプリケーションがメッセージを送信すると、メッセージは失われます。アプリケーションは、メッセージを送信する前にクライアントの接続状態を確認する必要があります。

クライアント側

ウェブクライアントアプリケーションに次のコードを記述します。

1. アプリケーションから受信メッセージを受信するコールバック関数を定義します。

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

2. `clientConnection.applicationMessage` をコールバック関数に設定します。

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

3. `GameLiftStreams.sendApplicationMessage` 関数を呼び出して、アプリケーションにメッセージを送信します。これは、ストリームセッションがアクティブで入力がアタッチされている限り、いつでも呼び出すことができます。

例として、Amazon GameLift Streams Web SDK サンプルクライアントを参照してください。このクライアントは、クライアント側でシンプルなデータチャネルを設定する方法を示しています。

アプリケーション側

アプリケーションに次のロジックを記述します。

ステップ 1. データチャネルポートに接続する

アプリケーションが起動したら、40712のポートに接続しますlocalhost。アプリケーションは、実行期間中、この接続を維持する必要があります。アプリケーションが接続を閉じると、再度開くことはできません。

ステップ 2. イベントをリッスンする

イベントは固定サイズのヘッダーで始まり、その後に可変長の関連データが続きます。アプリケーションがイベントを受け取ったら、イベントを解析して情報を取得します。

Event format

- ヘッダー: 形式の 4 バイトヘッダー abcc
 - a : クライアント ID バイト。これは、複数の接続(切断と再接続による)の場合に、特定のクライアント接続を識別します。
 - b : イベントタイプのバイト。 -0 接続されたクライアント、 -1 接続されていないクライアント、 2 - クライアントからメッセージが送信されます。その他のイベントタイプは、今後の Amazon GameLift Streams サービスの更新で受信される可能性があるため、無視する必要があります。
 - cc : 関連付けられたイベントデータの長さ。これは、ビッグエンディアンの順序で 2 バイトとして表されます(最初のバイトが最も重要です)。イベントタイプが 2 の場合、イベントデータはクライアントからのメッセージの内容を表します。
- データ: 残りのバイトには、クライアントメッセージなどのイベントデータが含まれます。データの長さは、ヘッダーcc ので示されます。

イベントをリッスンするには

1. 4 つのヘッダーバイトを読み取って、クライアント ID、イベントタイプ、イベントデータの長さを取得します。
2. ヘッダーで説明されている長さに従って、クライアント ID とイベントタイプに関係なく、可変長イベントデータを読み取ります。イベントデータがバッファに残されないように、データを無条件に読み取ることが重要です。バッファには、次のイベントヘッダーと混同される可能性があります。イベントタイプに基づいてデータの長さを推測しないでください。
3. アプリケーションで認識された場合は、イベントタイプに基づいて適切なアクションを実行します。このアクションには、着信接続または切断のログ記録、クライアントメッセージの解析、アプリケーションロジックのトリガーなどがあります。

ステップ 3. クライアントにメッセージを送信する

アプリケーションは、受信イベントで使用されるのと同じ 4 バイトのヘッダー形式でメッセージを送信する必要があります。

クライアントにメッセージを送信するには

1. 次のプロパティを使用してヘッダーを書き込みます。

- a : クライアント ID バイト。メッセージがクライアントメッセージに応答している場合は、古いクライアント接続から新しく再接続されたクライアントに応答を配信するなどの競合状態を避けるため、受信クライアントメッセージと同じクライアント ID を再利用する必要があります。アプリケーションが未承諾メッセージをクライアントに送信している場合は、最新の「クライアント接続」イベント (イベントタイプ 0) と一致するようにクライアント ID を設定する必要があります。
 - b : 送信メッセージのイベントタイプは常に 2 である必要があります。クライアントは、他のイベントタイプのメッセージを無視します。
 - c. cc : メッセージの長さ、バイト単位。
2. メッセージバイトを書き込みます。

クライアントが切断しない限り、メッセージは指定されたクライアントに配信されます。切断されたクライアントが再接続すると、クライアント接続イベントを介して新しいクライアント ID が割り当てられます。古いクライアント ID の未配信メッセージはすべて破棄されます。

Example

次の擬似コードは、アプリケーション側でメッセージを伝えるロジックを示しています。Winsockを使用した完全な例については、Windows Sockets 2 ドキュメントの[「Winsock クライアントコードの完了」](#)を参照してください。

```
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)
```

Amazon GameLift Streams 起動チェックリスト

Amazon GameLift Streams の正常な起動の準備には、計画と調整が必要です。この詳細なチェックリストに従って、イベントまでの数週間でスムーズなエクスペリエンスを確保してください。

Amazon GameLift Streams チームに通知する

アクション: 少なくとも 8 週間前に、テクニカルアカウントマネージャー、アカウントチーム、またはアカウントソリューションアーキテクトに、起動タイムラインと予想されるピーク同時ストリームについて通知します。

理由: 本番ワークフローの規模を理解することは、サービスの制限が適切であることを確認し、必要に応じて調整するのに役立ちます。また、容量の可用性に関するガイダンスと、起動に関する推奨事項も提供します。

互換性とパフォーマンスのテスト

アクション: アプリケーションを大規模かつキャパシティのあるすべての場所でテストし、ポジティブなカスタマーエクスペリエンスを確認します。Amazon GameLift Streams は、さまざまなレベルのパフォーマンスとランタイムをサポートする NVIDIA ベースのストリームクラスを提供します。

理由: 徹底的なテストは、起動前に潜在的な互換性とパフォーマンスの問題を特定して解決するのに役立ちます。ストリームクラスについては、次の点に注意してください。

- 「高」ストリームクラスはマルチテナントをサポートしているため、2 つのアプリケーションを 1 つのインスタンスで同時に実行できます。「高」ストリームクラスを使用している場合は、少なくとも 2 つの同時ストリームでテストして、CPU、GPU、メモリなどの共有リソースでアプリケーションがどのように動作するかを確認します。

キャパシティ予約

アクション: 起動の少なくとも 8 週間前に、特に重要で大規模なニーズが予想される場合は、アカウントチームに連絡してキャパシティを予約してください。互換性テスト、パフォーマンス要件、予算に基づいて、ストリームクラスとストリーミング場所を決定します。開始/終了時刻と必要なキャパ

シティーを指定します。AWS では、すべてのキャパシティー予約を、予約の必要期限の 6~8 週間前に確定する必要があります。

理由: Amazon GameLift Streams は、オンデマンド容量を使用して先着順で動作します。予約は、必要な容量を保証するために不可欠です。

大規模なパフォーマンステスト

アクション: APIs と Amazon GameLift Streams 設定を徹底的に負荷テストして、負荷時のパフォーマンス(レイテンシー、解像度、フレームレート)を確認します。必ず [Amazon GameLift Streams API レート制限](#) をチェックして、起動以降に必要な十分なヘッドルームがあることを確認してください。制限の引き上げが必要になると思われる場合は、アカウントマネージャーに問い合わせるか、サポートチケットを送信してください。

理由: ロードテストでは、アプリケーションと Amazon GameLift Streams 設定が起動前にストレスの下でどのように動作するかが明らかになります。これは、大規模なパフォーマンスをスムーズにするために不可欠です。

起動前セットアップ

アクション: 起動の少なくとも 2~3 日前に、最終的なアプリケーションリソースとストリームグループを作成します。ストリーミングパフォーマンスを検証し、必要に応じて容量をスケールアップします。

理由: これにより、すべてのコンポーネントが期待どおりに動作し、予期しない問題のリスクを最小限に抑え、イベント中の診断と復旧が容易になります。

その他のヒント

- 整合性が重要です。起動イベント全体で同じ既存のストリームグループを使用すると、Amazon GameLift Streams バックエンドの整合性が維持され、トラブルシューティングが簡素化されます。
- 綿密にモニタリングする: パフォーマンスとユーザーのフィードバックを注意深くモニタリングして、問題に迅速に対処します。運用ダッシュボードを構築します。Amazon CloudWatch を使用してストリームの容量、使用状況、パフォーマンス [CloudWatch を使用して監視する](#) をモニタリングします(詳細については、「」を参照)。その他のガイダンスについては、[Well-Architected フレームワーク](#) を参照してください。

さらにサポートが必要ですか？

ご質問がある場合やサポートが必要な場合は、[Amazon GameLift Streams サポート](#)までお問い合わせください。お客様の起動が成功し、シームレスになるようにお手伝いします。

Amazon GameLift Streams のセキュリティ

でのクラウドセキュリティが最優先事項 AWS です。お客様は AWS、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャを活用できます。

セキュリティは、AWS お客様とお客様の間の責任共有です。[責任共有モデル](#)ではこれをクラウドのセキュリティおよびクラウド内のセキュリティと説明しています。

- クラウドのセキュリティ – AWS は、AWS サービスを実行するインフラストラクチャを保護する責任があります AWS クラウド。AWS また、では、安全に使用できるサービスも提供しています。サードパーティーの監査者は、[AWS コンプライアンスプログラム](#)コンプライアンスプログラムの一環として、当社のセキュリティの有効性を定期的にテストおよび検証。Amazon GameLift Streams に適用されるコンプライアンスプログラムの詳細については、「コンプライアンスプログラム[AWS による対象範囲内のサービスコンプライアンスプログラム](#)」を参照してください。
- クラウドのセキュリティ – お客様の責任は、使用する AWS サービスによって決まります。Amazon GameLift Streams は、ユーザーが提供するプログラムを実行するように設計されており、ユーザーはそれらのプログラムの内容とセキュリティについて単独で責任を負います。また、ユーザーは、データの機密性、会社の要件、適用される法律や規制など、その他の要因についても責任を負います。

このドキュメントは、Amazon GameLift Streams を使用する際の責任共有モデルの適用方法を理解するのに役立ちます。以下のトピックでは、セキュリティおよびコンプライアンスの目的を達成するように Amazon GameLift Streams を設定する方法について説明します。また、Amazon GameLift Streams リソースのモニタリングや保護に役立つ他の AWS サービスの使用方法についても説明します。

トピック

- [Amazon GameLift Streams でのデータ保護](#)
- [Amazon GameLift Streams の Identity and Access Management](#)
- [Amazon GameLift Streams のコンプライアンス検証](#)
- [Amazon GameLift Streams の耐障害性](#)
- [Amazon GameLift Streams のインフラストラクチャセキュリティ](#)
- [Amazon GameLift Streams の設定と脆弱性の分析](#)
- [Amazon GameLift Streams のセキュリティのベストプラクティス](#)

Amazon GameLift Streams でのデータ保護

責任 AWS [共有モデル](#)、Amazon GameLift Streams でのデータ保護に適用されます。このモデルで説明されているように、AWS はすべての を実行するグローバルインフラストラクチャを保護する責任があります AWS クラウド。ユーザーは、このインフラストラクチャでホストされるコンテンツに対する管理を維持する責任があります。また、使用する「AWS のサービス」のセキュリティ設定と管理タスクもユーザーの責任となります。データプライバシーの詳細については、[データプライバシーに関するよくある質問](#)を参照してください。欧州でのデータ保護の詳細については、AWS セキュリティブログに投稿された「[AWS 責任共有モデルおよび GDPR](#)」のブログ記事を参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント、AWS IAM アイデンティティセンターまたは AWS Identity and Access Management (IAM) を使用して個々のユーザーを設定することをお勧めします。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします：

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 は必須ですが、TLS 1.3 を推奨します。
- で API とユーザーアクティビティのログ記録を設定します AWS CloudTrail。CloudTrail 証跡を使用して AWS アクティビティをキャプチャする方法については、「AWS CloudTrail ユーザーガイド」の[CloudTrail 証跡の使用](#)」を参照してください。
- AWS 暗号化ソリューションと、その中のすべてのデフォルトのセキュリティコントロールを使用します AWS のサービス。
- Amazon Macie などの高度な管理されたセキュリティサービスを使用します。これらは、Amazon S3 に保存されている機密データの検出と保護を支援します。
- コマンドラインインターフェイスまたは API AWS を介してにアクセスするときに FIPS 140-3 検証済み暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの詳細については、「[連邦情報処理規格 \(FIPS\) 140-3](#)」を参照してください。

お客様の E メールアドレスなどの極秘または機密情報を、タグ、または [名前] フィールドなどの自由形式のテキストフィールドに含めないことを強くお勧めします。これは、コンソール、API、AWS CLI または SDK を使用して Amazon GameLift Streams または他の AWS のサービスを操作する場合も同様です。AWS SDKs タグ、または名前に使用される自由記述のテキストフィールドに入力したデータは、請求または診断ログに使用される場合があります。外部サーバーに URL を提供す

る場合、そのサーバーへのリクエストを検証できるように、認証情報を URL に含めないことを強くお勧めします。

Amazon GameLift Streams は、サービス固有のデータを次のように処理します。

- お客様が用意したアプリケーション – Amazon GameLift Streams は、提供されている場合、内部サービススマネージド Amazon S3 バケットと Amazon EC2 インスタンスにアタッチされた NVME ストレージドライブに顧客データを保存します。すべてのデータは、サービススマネージド型の保管時の暗号化で保存されます。お客様がこのデータのコピーに直接アクセスすることはできません。アプリケーションを削除するには、Amazon GameLift Streams コンソールまたはサービス API を使用します。
- お客様が用意したメタデータ – お客様は、説明、接続情報、顧客 IDs などの不透明な識別子を含むメタデータを Amazon GameLift Streams APIs に提供できます。このメタデータは常に特定のカスタマーリソースに関連付けられます。
- お客様が生成したデータ – アプリケーションが通常のオペレーションの一部として新しいデータを書き込む場合、このお客様が生成したデータはユーザーセッションの終了まで保持されます。セッションの終了時に、生成されたデータは、オプションでお客様が選択した Amazon S3 バケットの宛先にエクスポートできます。それ以外の場合、お客様が生成したデータは、生成された Amazon EC2 インスタンスを残しません。データ処理の詳細については、[セッション分離](#)に関するトピックを参照してください。
- メトリクスとイベントデータ – Amazon GameLift Streams メトリクスとイベントデータ。Amazon GameLift Streams コンソールまたは サービス API の呼び出しからアクセスできます。データは、アプリケーション、ストリームグループ、ストリームセッションで使用できます。承認されたユーザーは、Amazon CloudWatch および CloudWatch Events を通じてこのデータにアクセスすることもできます。

 **Important**

Amazon GameLift Streams に顧客 IDs またはその他の識別子を提供する場合、これらの値は匿名化された参照であり、機密情報や個人情報が含まれていないことが予想されます。Amazon GameLift Streams はメタデータフィールドを編集しません。

データ保護の詳細については、AWS セキュリティブログのブログ投稿「[AWS の責任共有モデルと GDPR](#)」を参照してください。

保管中の暗号化

Amazon GameLift Streams 固有のデータの保管時の暗号化は、次のように処理されます。

- ・ アプリケーションコンテンツは、サービスマネージドの暗号化された Amazon S3 バケット、さらにサービスマネージドの Amazon EC2 インスタンスにアタッチされたハードウェア暗号化された NVME ドライブに保存されます。

転送中の暗号化

Amazon GameLift Streams APIs の呼び出しは、安全な (SSL) 接続を介して行われ、[AWS 署名バージョン 4](#) を使用して認証されます (CLI または AWS SDK AWS を介して接続すると、署名は自動的に処理されます)。呼び出しエンティティは、Amazon GameLift Streams リソースに定義されている IAM アクセスポリシーを適用して認証されるセキュリティ認証情報を使用します。

マルチロケーションストリームグループのコンテキストでは、ストリーミング容量が割り当てられたストリームグループ内の任意の場所からアプリケーションをストリーミングするために、Amazon GameLift Streams はそれらの場所にアプリケーションを安全にレプリケートします。

同様に、Amazon GameLift Streams は、リクエストに応じて、ログデータとセッションファイルを、セッションの終了時にお客様が名前を付けた Amazon S3 バケットに保存します。バケットがセッションと同じ場所にない場合、Amazon GameLift Streams はバケット AWS リージョンがあるにファイルを安全に転送します。

エンドユーチャー ストリームの保護

個々のエンドユーチャー ストリームは、エンドユーザーのウェブブラウザと Amazon GameLift Streams バックエンドホスト間の直接接続です。これらのストリームは業界標準の WebRTC 暗号化で保護され、ストリームの両方のエンドポイントは、ストリームセッション API を介してネゴシエートされた `SignalRequest` および `SignalResponse` 値の一部である暗号化識別子によって正に識別されます。APIs

データチャネルメッセージは、ストリームに使用される WebRTC 暗号化によってもカバーされます。これらのメッセージは Amazon GameLift Streams によって復号され、暗号化されていない API を介してお客様のアプリケーションに on-the-host ポートカルに渡されます。Amazon GameLift Streams サービスでも復号できない end-to-end の暗号化が必要な場合、この追加の暗号化レイヤーはアプリケーション開発者の責任となります。

Linux ストリームクラスのセッション分離

Linux ストリームクラス (Ubuntu および Proton ランタイム) では、Amazon GameLift Streams はコンテナ分離を使用します。すべてのセッションは新しい Linux コンテナで実行され、使用後に破棄されます。つまり、新しい各セッションは、コンピューティングリソースを共有する他のユーザーから分離された新しい環境で実行されます（共有リソースストリームクラスで実行されている場合）。新しいセッションの起動時に、以前のセッションのデータは存在しません。

Windows ストリームクラスのセッション分離

Windows ストリームクラス (Microsoft Windows Server ランタイム) では、Amazon GameLift Streams はソフトウェア分離を使用します。このサービスは、セッション間の重要なシステム状態をリセットするためにソフトウェアエージェントに依存します。一部のフォルダは、ホスト上のディスクキャッシュなど、パフォーマンスを最適化するために複数のセッションにわたって保持されます。ソフトウェアエージェントは、前のストリームセッション中にユーザーのプロファイルディレクトリで生成されたファイルを自動的に削除します。ただし、エージェントは、アプリケーションの実行前に存在し、アプリケーションの実行中に変更されたファイルを削除しません。また、アプリケーションが追加した Windows レジストリキーも削除されません。お客様は、オペレーティングシステム全体の整合性を損なうことを避ける責任があることに注意してください。アプリケーションは管理者ユーザーとして実行され、複数のセッションにまたがる変更など、重要なシステムレベルのファイルへの変更を許可する場合があります。アプリケーションを保護し、安全でない、または不安定なオペレーティングシステムの変更が発生しないようにするのはお客様の責任です。

お客様は、アプリケーションの起動時に、変更されたファイルと以前のセッションから追加されたレジストリキーをクリーンアップする責任があります。これは、アプリケーションがユーザーのプロファイルディレクトリに書き込む機密情報を保護するための重要なステップです。そのために、お客様は以下のアクションを実行する独自のカスタムスクリプトを作成できます。

- ・ アプリケーションによって変更された%USERPROFILE%ディレクトリ外のファイルを復元します。
- ・ アプリケーションが追加した機密またはユーザー固有のレジストリキーをクリーンアップします。

暗号化キーの管理

サービスは AWS マネージド暗号化キーを使用します。各リージョンは個別の KMS キーを使用します。カスタマーマネージドキー (CMKs) はサポートされていません。

Amazon GameLift Streams に提供されたアプリケーションファイルは、サービスから再発行またはエクスポートできません。お客様は、サービスコンソールまたは APIs を使用してアプリケーション

を削除できます。以前にこれらのアプリケーションファイルを保持していたドライブは、関連付けられたストリームグループを削除することで完全に消去できます。

ネットワーク間トラフィックのプライバシー

Amazon GameLift Streams は、パブリック向けネットワークを使用してストリームセッションをホストします。各ストリームグループは、他のストリームグループや他の顧客から分離された 1 つ以上のサービスマネージド VPC ネットワークで構成されます。インバウンドネットワーク接続は、認証されたサービスプローラー WebRTC ストリーム接続を除いて拒否されます。お客様のアプリケーションは、制限なしにこれらの VPCs から他のパブリックアドレスに接続できます。

さらに、お客様がサービス API コールまたは設定のみを使用してストリームまたはアプリケーションデータをパブリックにアクセス可能にする方法はありません。すべてのサービスインターフェンスは、認証された API AWS コールによってゲートされます。顧客がストリームをパブリックにアクセスできるようにする場合は、独自のクライアントウェブアプリケーションを作成する必要があります。これにより、認証された呼び出しを実行してストリームを開始して表示できます。

Amazon GameLift Streams の Identity and Access Management

AWS Identity and Access Management (IAM) は、管理者が AWS リソースへのアクセスを安全に制御 AWS のサービスするのに役立つです。IAM 管理者は、誰を認証 (サインイン) し、誰に Amazon GameLift Streams リソースの使用を許可する (アクセス許可を付与する) かを制御します。IAM は、追加料金なしで使用できる AWS のサービスです。

トピック

- [オーディエンス](#)
- [アイデンティティを使用した認証](#)
- [ポリシーを使用したアクセスの管理](#)
- [Amazon GameLift Streams と IAM の連携方法](#)
- [Amazon GameLift Streams のアイデンティティベースのポリシーの例](#)
- [Amazon GameLift Streams のアイデンティティとアクセスのトラブルシューティング](#)

オーディエンス

AWS Identity and Access Management (IAM) の使用方法は、ロールによって異なります。

- ・ サービスユーザー - 機能にアクセスできない場合は、管理者にアクセス許可をリクエストします（「[Amazon GameLift Streams のアイデンティティとアクセスのトラブルシューティング](#)」を参照）。
- ・ サービス管理者 - ユーザーアクセスを決定し、アクセス許可リクエストを送信します（「[Amazon GameLift Streams と IAM の連携方法](#)」を参照）
- ・ IAM 管理者 - アクセスを管理するためのポリシーを作成します（「[Amazon GameLift Streams のアイデンティティベースのポリシーの例](#)」を参照）

アイデンティティを使用した認証

認証とは、ID 認証情報 AWS を使用して にサインインする方法です。、IAM ユーザー AWS アカウントのルートユーザー、または IAM ロールを引き受けることで認証される必要があります。

AWS IAM アイデンティティセンター (IAM Identity Center)、シングルサインオン認証、Google/Facebook 認証情報などの ID ソースからの認証情報を使用して、フェデレーテッド ID としてサインインできます。サインインの詳細については、「AWS サインイン ユーザーガイド」の「[AWS アカウントにサインインする方法](#)」を参照してください。

プログラムによるアクセスの場合、は SDK と CLI AWS を提供してリクエストを暗号化して署名します。詳細については、「IAM ユーザーガイド」の「[API リクエストに対するAWS 署名バージョン4](#)」を参照してください。

AWS アカウント ルートユーザー

を作成するときは AWS アカウント、すべての AWS のサービス および リソースへの完全なアクセス権を持つ AWS アカウント ルートユーザーと呼ばれる 1 つのサインインアイデンティティから始めます。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザー認証情報を必要とするタスクについては、「IAM ユーザーガイド」の「[ルートユーザー認証情報が必要なタスク](#)」を参照してください。

フェデレーテッドアイデンティティ

ベストプラクティスとして、人間のユーザーは ID プロバイダーとのフェデレーションを使用して、一時的な認証情報 AWS のサービス を使用して にアクセスする必要があります。

フェデレーテッド ID は、エンタープライズディレクトリ、ウェブ ID プロバイダー、または ID ソースの認証情報 AWS のサービス を使用して Directory Service にアクセスするユーザーです。フェデレーテッドアイデンティティは、一時的な認証情報を提供するロールを受けます。

アクセスを一元管理する場合は、AWS IAM アイデンティティセンターをお勧めします。詳細については、「AWS IAM アイデンティティセンター ユーザーガイド」の「[IAM アイデンティティセンターとは](#)」を参照してください。

IAM ユーザーとグループ

[IAM ユーザー](#)は、特定の個人やアプリケーションに対する特定のアクセス許可を持つアイデンティティです。長期認証情報を持つ IAM ユーザーの代わりに一時的な認証情報を使用することをお勧めします。詳細については、IAM ユーザーガイドの「[ID プロバイダーとのフェデレーションを使用してにアクセスする必要がある AWS](#)」を参照してください。

[IAM グループ](#)は、IAM ユーザーの集合を指定し、大量のユーザーに対するアクセス許可の管理を容易にします。詳細については、「IAM ユーザーガイド」の「[IAM ユーザーに関するユースケース](#)」を参照してください。

IAM ロール

[IAM ロール](#)は、特定のアクセス許可を持つアイデンティであり、一時的な認証情報を提供します。ユーザーから IAM ロール (コンソール) に切り替えるか、または API オペレーションを呼び出すことで、ロールを引き受けることができます。AWS CLI AWS 詳細については、「IAM ユーザーガイド」の「[ロールを引き受けるための各種方法](#)」を参照してください。

IAM ロールは、フェデレーションユーザーアクセス、一時的な IAM ユーザーのアクセス許可、クロスアカウントアクセス、クロスサービスアクセス、および Amazon EC2 で実行するアプリケーションに役立ちます。詳細については、IAM ユーザーガイド の「[IAM でのクロスアカウントリソースアクセス](#)」を参照してください。

ポリシーを使用したアクセスの管理

でアクセスを制御する AWS には、ポリシーを作成し、ID AWS またはリソースにアタッチします。ポリシーは、アイデンティティまたはリソースに関連付けられたときにアクセス許可を定義します。は、プリンシパルがリクエストを行うときにこれらのポリシー AWS を評価します。ほとんどのポリシーは JSON ドキュメント AWS として に保存されます。JSON ポリシードキュメントの詳細については、「IAM ユーザーガイド」の「[JSON ポリシー概要](#)」を参照してください。

管理者は、ポリシーを使用して、どのプリンシパルがどのリソースに対して、どのような条件でアクションを実行できるかを定義することで、誰が何にアクセスできるかを指定します。

デフォルトでは、ユーザーやロールにアクセス許可はありません。IAM 管理者は IAM ポリシーを作成してロールに追加し、このロールをユーザーが引き受けられるようにします。IAM ポリシーは、オペレーションの実行方法を問わず、アクセス許可を定義します。

アイデンティティベースのポリシー

アイデンティティベースのポリシーは、アイデンティティ（ユーザー、グループ、またはロール）にアタッチできる JSON アクセス許可ポリシードキュメントです。これらのポリシーは、アイデンティティがどのリソースに対してどのような条件下でどのようなアクションを実行できるかを制御します。アイデンティティベースポリシーの作成方法については、IAM ユーザーガイドの [カスタマーマネジメントポリシーでカスタム IAM アクセス許可を定義する](#) を参照してください。

アイデンティティベースのポリシーは、オンラインポリシー（单一の ID に直接埋め込む）または管理ポリシー（複数の ID にアタッチされたスタンダードアロンポリシー）にすることができます。管理ポリシーとオンラインポリシーのいずれかを選択する方法については、「IAM ユーザーガイド」の「[管理ポリシーとオンラインポリシーのいずれかを選択する](#)」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。例としては、IAM ロール信頼ポリシーや Amazon S3 バケットポリシーなどがあります。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。リソースベースのポリシーでは、[プリンシパルを指定する必要があります](#)。

リソースベースのポリシーは、そのサービス内にあるオンラインポリシーです。リソースベースのポリシーでは、IAM の AWS マネージドポリシーを使用できません。

その他のポリシータイプ

AWS は、より一般的なポリシータイプによって付与されるアクセス許可の上限を設定できる追加のポリシータイプをサポートしています。

- アクセス許可の境界 – アイデンティティベースのポリシーで IAM エンティティに付与することのできるアクセス許可の数の上限を設定します。詳細については、「IAM ユーザーガイド」の「[IAM エンティティのアクセス許可境界](#)」を参照してください。
- サービスコントロールポリシー (SCP) - AWS Organizations内の組織または組織単位の最大のアクセス許可を指定します。詳細については、「AWS Organizations ユーザーガイド」の「[サービスコントロールポリシー](#)」を参照してください。

- リソースコントロールポリシー (RCP) – は、アカウント内のリソースで利用できる最大数のアクセス許可を定義します。詳細については、「AWS Organizations ユーザーガイド」の「[リソースコントロールポリシー \(RCP\)](#)」を参照してください。
- セッションポリシー – ロールまたはフェデレーションユーザーの一時セッションを作成する際にパラメータとして渡される高度なポリシーです。詳細については、「IAM ユーザーガイド」の「[セッションポリシー](#)」を参照してください。

複数のポリシータイプ

1つのリクエストに複数のタイプのポリシーが適用されると、結果として作成されるアクセス許可を理解するのがさらに難しくなります。が複数のポリシータイプが関与する場合にリクエストを許可するかどうか AWS を決定する方法については、「IAM ユーザーガイド」の「[ポリシー評価ロジック](#)」を参照してください。

Amazon GameLift Streams と IAM の連携方法

IAM を使用して Amazon GameLift Streams へのアクセスを管理する前に、Amazon GameLift Streams で使用できる IAM 機能を確認してください。

Amazon GameLift Streams で使用できる IAM 機能

IAM の特徴量	Amazon GameLift Streams のサポート
アイデンティティベースのポリシー	あり
リソースベースのポリシー	なし
ポリシーアクション	あり
ポリシーリソース	はい
ポリシー条件キー (サービス固有)	はい
ACL	なし
ABAC (ポリシー内のタグ)	一部。 ABAC は、アプリケーションとストリームグループでのみサポートされています。
一時認証情報	あり

IAM の特徴量	Amazon GameLift Streams のサポート
<u>プリンシパルアクセス権限</u>	あり
<u>サービスロール</u>	いいえ
<u>サービスリンクロール</u>	不可

Amazon GameLift Streams およびその他の AWS のサービスがほとんどの IAM 機能と連携する方法の概要については、「IAM ユーザーガイド」の[AWS 「IAM と連携する のサービス」](#)を参照してください。

Amazon GameLift Streams のアイデンティティベースのポリシー

アイデンティティベースのポリシーのサポート: あり

アイデンティティベースポリシーは、IAM ユーザー、ユーザーグループ、ロールなど、アイデンティティにアタッチできる JSON 許可ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。アイデンティティベースポリシーの作成方法については、「IAM ユーザーガイド」の「[カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する](#)」を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。JSON ポリシーで使用できるすべての要素について学ぶには、「IAM ユーザーガイド」の「[IAM JSON ポリシーの要素のリファレンス](#)」を参照してください。

Amazon GameLift Streams のアイデンティティベースのポリシーの例

Amazon GameLift Streams アイデンティティベースのポリシーの例を表示するには、「」を参照してください[Amazon GameLift Streams のアイデンティティベースのポリシーの例](#)。

Amazon GameLift Streams 内のリソースベースのポリシー

リソースベースのポリシーのサポート: なし

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使

用して特定のリソースへのアクセスをコントロールできます。ポリシーがアタッチされているリソースの場合、指定されたプリンシバルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーで、[プリンシバルを指定する](#)必要があります。プリンシバルには、アカウント、ユーザー、ロール、フェデレーティッドユーザー、または を含めることができます AWS のサービス。

クロスアカウントアクセスを有効にするには、全体のアカウント、または別のアカウントの IAM インティティを、リソースベースのポリシーのプリンシバルとして指定します。詳細については、IAM ユーザーガイドの[IAM でのクロスアカウントリソースアクセス](#)を参照してください。

Amazon GameLift Streams のポリシーアクション

ポリシーアクションのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシバルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

JSON ポリシーの Action 要素にはポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。このアクションは関連付けられたオペレーションを実行するためのアクセス許可を付与するポリシーで使用されます。

Amazon GameLift Streams のポリシーアクションは、アクションの前に次のプレフィックスを使用します。

gameliftstreams

単一のステートメントで複数のアクションを指定するには、アクションをカンマで区切ります。

Example

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

Amazon GameLift Streams アイデンティティベースのポリシーの例を表示するには、「」を参照してください[Amazon GameLift Streams のアイデンティティベースのポリシーの例](#)。

Amazon GameLift Streams のポリシーリソース

ポリシーリソースのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Resource JSON ポリシー要素はアクションが適用されるオブジェクトを指定します。ベストプラクティスとして、[Amazon リソースネーム \(ARN\)](#) を使用してリソースを指定します。リソースレベルのアクセス許可をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

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

Amazon GameLift Streams アイデンティティベースのポリシーの例を表示するには、「」を参照してください[Amazon GameLift Streams のアイデンティティベースのポリシーの例](#)。

Amazon GameLift Streams のポリシー条件キー

サービス固有のポリシー条件キーのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Condition 要素は、定義された基準に基づいてステートメントが実行される時期を指定します。イコールや未満などの条件演算子を使用して条件式を作成して、ポリシーの条件とリクエスト内の値を一致させることができます。すべての AWS グローバル条件キーを確認するには、「IAM ユーザーガイド」の[AWS 「グローバル条件コンテキストキー」](#)を参照してください。

Amazon GameLift Streams アイデンティティベースのポリシーの例を表示するには、「」を参照してください[Amazon GameLift Streams のアイデンティティベースのポリシーの例](#)。

Amazon GameLift Streams ACLs

ACL のサポート: なし

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするためのアクセス許可を持つかを制御します。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

Amazon GameLift Streams での ABAC

ABAC (ポリシー内のタグ) のサポート: 一部

属性ベースのアクセスコントロール (ABAC) は、タグと呼ばれる属性に基づいてアクセス許可を定義する認可戦略です。IAM エンティティと AWS リソースにタグをアタッチし、プリンシパルのタグがリソースのタグと一致するときにオペレーションを許可するように ABAC ポリシーを設計できます。

タグに基づいてアクセスを管理するには、`aws:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの条件要素でタグ情報を提供します。

サービスがすべてのリソースタイプに対して 3 つの条件キーすべてをサポートする場合、そのサービスの値はあります。サービスが一部のリソースタイプに対してのみ 3 つの条件キーのすべてをサポートする場合、値は「部分的」になります。

ABAC の詳細については、「IAM ユーザーガイド」の「[ABAC 認可でアクセス許可を定義する](#)」を参照してください。ABAC をセットアップする手順を説明するチュートリアルについては、「IAM ユーザーガイド」の「[属性ベースのアクセスコントロール \(ABAC\) を使用する](#)」を参照してください。

Amazon GameLift Streams での一時的な認証情報の使用

一時的な認証情報のサポート: あり

一時的な認証情報は、AWS リソースへの短期的なアクセスを提供し、フェデレーションまたは切り替えロールを使用する場合に自動的に作成されます。長期的なアクセスキーを使用する代わりに、一時的な認証情報を動的に生成 AWS することをお勧めします。詳細については、「IAM ユーザーガイド」の「[IAM の一時的な認証情報](#)」および「[AWS のサービスと IAMとの連携](#)」を参照してください。

Amazon GameLift Streams のクロスサービスプリンシパルアクセス許可

転送アクセスセッション (FAS) のサポート: あり

転送アクセスセッション (FAS) は、[呼び出すプリンシバルのアクセス許可と AWS のサービス、ダウンストリームサービス AWS のサービスへのリクエストをリクエストする](#)を使用します。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。

新しいアプリケーションリソースを作成する場合、Amazon GameLift Streams は呼び出し元プリンシバルのアクセス許可を使用して、顧客のアプリケーションファイルを含む Amazon S3 バケットにアクセスします。Amazon GameLift Streams は、呼び出し元プリンシバルを調べて、マルチロケーションストリームグループなどの特定のクロスリージョン機能のオプトイン適格性を検証します。

Amazon GameLift Streams のサービスロール

サービスロールのサポート: なし

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#)です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、IAM ユーザーガイドの [AWS のサービスに許可を委任するロールを作成する](#)を参照してください。

Warning

サービスロールのアクセス許可を変更すると、Amazon GameLift Streams の機能が破損する可能性があります。Amazon GameLift Streams が指示する場合にのみ、サービスロールを編集します。

Amazon GameLift Streams のサービスにリンクされたロール

サービスにリンクされたロールのサポート: なし

サービスにリンクされたロールは、[にリンクされたサービスロールの一覧](#)で AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは [に表示され AWS アカウント、サービスによって所有されます](#)。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。

サービスにリンクされたロールの作成または管理の詳細については、「[IAM と提携する AWS のサービス](#)」を参照してください。表の「サービスリンクロール」列に Yes と記載されたサービスを見つけます。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、[はい] リンクを選択します。

Amazon GameLift Streams のアイデンティティベースのポリシーの例

デフォルトでは、ユーザーとロールには Amazon GameLift Streams リソースを作成または変更するアクセス許可はありません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。

これらのサンプルの JSON ポリシードキュメントを使用して IAM アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「[IAM ポリシーを作成する \(コンソール\)](#)」を参照してください。

各リソースタイプの ARN の形式など、Amazon GameLift Streams で定義されるアクションとリソースタイプの詳細については、「サービス認可リファレンス」の「[Amazon GameLift Streams のアクション、リソース、および条件キー](#)」を参照してください。ARNs

トピック

- [ポリシーに関するベストプラクティス](#)
- [Amazon GameLift Streams コンソールの使用](#)
- [自分の権限の表示をユーザーに許可する](#)

ポリシーに関するベストプラクティス

ID ベースのポリシーは、アカウント内で誰かが Amazon GameLift Streams リソースを作成、アクセス、または削除できるかどうかを決定します。これらのアクションでは、AWS アカウントに費用が発生する場合があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください：

- AWS 管理ポリシーを開始し、最小特権のアクセス許可に移行 – ユーザーとワークロードにアクセス許可の付与を開始するには、多くの一般的なユースケースにアクセス許可を付与する AWS 管理ポリシーを使用します。これらはで使用できます AWS アカウント。ユースケースに固有の AWS カスタマー管理ポリシーを定義することで、アクセス許可をさらに減らすことをお勧めします。 詳細については、IAM ユーザーガイドの [AWS マネージドポリシー](#) または [ジョブ機能のAWS マネージドポリシー](#) を参照してください。
- 最小特権を適用する – IAM ポリシーでアクセス許可を設定する場合は、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権アクセス許可とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、IAM ユーザーガイドの [IAM でのポリシーとアクセス許可](#) を参照してください。

- IAM ポリシーで条件を使用してアクセスをさらに制限する - ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。たとえば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。条件を使用して、サービスアクションがなどの特定のを通じて使用されている場合に AWS のサービス、サービスアクションへのアクセスを許可することもできます CloudFormation。詳細については、IAM ユーザーガイドの [IAM JSON ポリシー要素:条件](#) を参照してください。
- IAM アクセスアナライザーを使用して IAM ポリシーを検証し、安全で機能的な権限を確保する - IAM アクセスアナライザーは、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、IAM ユーザーガイドの [IAM Access Analyzer でポリシーを検証する](#) を参照してください。
- 多要素認証 (MFA) を要求する - で IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、MFA をオンにしてセキュリティを強化します。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、IAM ユーザーガイドの [MFA を使用した安全な API アクセス](#) を参照してください。

IAM でのベストプラクティスの詳細については、IAM ユーザーガイドの [IAM でのセキュリティのベストプラクティス](#) を参照してください。

Amazon GameLift Streams コンソールの使用

Amazon GameLift Streams コンソールにアクセスするには、最小限のアクセス許可のセットが必要です。これらのアクセス許可により、の Amazon GameLift Streams リソースの詳細を一覧表示および表示できます AWS アカウント。最小限必要な許可よりも制限が厳しいアイデンティティベースのポリシーを作成すると、そのポリシーを持つエンティティ (ユーザーまたはロール) に対してコンソールが意図したとおりに機能しません。

AWS CLI または AWS API のみを呼び出すユーザーには、最小限のコンソールアクセス許可を付与する必要はありません。代わりに、実行しようとしている API オペレーションに一致するアクションのみへのアクセスが許可されます。

自分の権限の表示をユーザーに許可する

この例では、ユーザーイデンティティにアタッチされたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーの作成方法を示します。このポリシーには、コンソールで、

または AWS CLI または AWS API を使用してプログラムでこのアクションを実行するアクセス許可が含まれています。

```
{  
    "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": [  
                "iam:GetGroupPolicy",  
                "iam:GetPolicyVersion",  
                "iam GetPolicy",  
                "iam>ListAttachedGroupPolicies",  
                "iam>ListGroupPolicies",  
                "iam>ListPolicyVersions",  
                "iam>ListPolicies",  
                "iam>ListUsers"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

Amazon GameLift Streams のアイデンティティとアクセスのトラブル シューーティング

以下の情報は、Amazon GameLift Streams と IAM の使用時に発生する可能性がある一般的な問題の診断と修正に役立ちます。

トピック

- [Amazon GameLift Streams でアクションを実行する権限がありません](#)
- [自分の以外のユーザーに Amazon GameLift Streams リソース AWS アカウントへのアクセスを許可したい](#)

Amazon GameLift Streams でアクションを実行する権限がありません

アクションを実行する権限がないというエラーが表示された場合は、そのアクションを実行できるようにポリシーを更新する必要があります。

次のエラー例は、mateojackson IAM ユーザーがコンソールを使用して、ある *my-example-widget* リソースに関する詳細情報を表示しようとしたことを想定して、その際に必要な gameliftstreams:*GetWidget* アクセス許可を持っていない場合に発生するものです。

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

この場合、gameliftstreams:*GetWidget* アクションを使用して *my-example-widget* リソースへのアクセスを許可するように、mateojackson ユーザーのポリシーを更新する必要があります。

サポートが必要な場合は、AWS 管理者にお問い合わせください。サインイン資格情報を提供した担当者が管理者です。

自分の以外のユーザーに Amazon GameLift Streams リソース AWS アカウントへの アクセスを許可したい

これは Amazon GameLift Streams ではできません。すべての API アクセスは、リソースを所有するアカウントに制限されます。代わりに、外部でコンテンツを共有するお客様は、自分のアカウントを使用して、Amazon GameLift Streams APIs を使用して他のユーザーに代わって新しいストリームセッションを開始し、適切な接続情報をそれらの外部ユーザーのウェブブラウザに転送する責任があります。

Amazon GameLift Streams のコンプライアンス検証

AWS のサービスが特定のコンプライアンスプログラムの範囲内にあるかどうかを確認するには、[AWS のサービス「コンプライアンスプログラムによる対象範囲内」](#) の「コンプライアンス」を参照し、関心のあるコンプライアンスプログラムを選択します。一般的な情報については、[AWS 「コンプライアンスプログラム」](#) を参照してください。

を使用して、サードパーティの監査レポートをダウンロードできます AWS Artifact。詳細については、[「Downloading Reports in AWS Artifact」](#) を参照してください。

を使用する際のお客様のコンプライアンス責任 AWS のサービスは、お客様のデータの機密性、貴社のコンプライアンス目的、適用される法律および規制によって決まります。を使用する際のコンプライアンス責任の詳細については AWS のサービス、[AWS 「セキュリティドキュメント」](#) を参照してください。

Amazon GameLift Streams の耐障害性

AWS グローバルインフラストラクチャは、AWS リージョン およびアベイラビリティーゾーンを中心に構築されています。は、低レイテンシー、高スループット、高度に冗長なネットワークで接続された、物理的に分離および分離された複数のアベイラビリティーゾーン AWS リージョン を提供します。アベイラビリティーゾーンでは、ゾーン間で中断することなく自動的にフェールオーバーするアプリケーションとデータベースを設計および運用することができます。アベイラビリティーゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性、フォールトトランク、および拡張性が優れています。

AWS リージョン およびアベイラビリティーゾーンの詳細については、[AWS 「グローバルインフラストラクチャ」](#) を参照してください。

AWS グローバルインフラストラクチャが提供するデータの冗長性に加えて、Amazon GameLift Streams は回復力のあるマルチアベイラビリティーゾーンインフラストラクチャで構築されています。アベイラビリティーゾーンが停止した場合、個々の既存のセッションが影響を受ける可能性がありますが、サービスは正常なアベイラビリティーゾーン間で新しいセッションの負荷分散を継続します。

Amazon GameLift Streams のインフラストラクチャセキュリティ

マネージドサービスである Amazon GameLift Streams は、AWS グローバルネットワークセキュリティで保護されています。AWS セキュリティサービスと インフラストラクチャ AWS を保護する方法については、[AWS 「クラウドセキュリティ」](#) を参照してください。インフラストラクチャ

セキュリティのベストプラクティスを使用して環境を AWS 設計するには、「Security Pillar AWS Well-Architected Framework」の [「Infrastructure Protection」](#) を参照してください。

AWS 公開された API コールを使用して、ネットワーク経由で Amazon GameLift Streams にアクセスします。クライアントは以下をサポートする必要があります。

- Transport Layer Security (TLS)。TLS 1.2 が必須で、TLS 1.3 をお勧めします。
- DHE (楕円ディフィー・ヘルマン鍵共有) や ECDHE (楕円曲線ディフィー・ヘルマン鍵共有) などの完全前方秘匿性 (PFS) による暗号スイート。これらのモードは Java 7 以降など、ほとんどの最新システムでサポートされています。

Amazon GameLift Streams での再利用とマルチテナンシー

Amazon GameLift Streams は、ストリームグループ間または他の AWS 顧客とコンピューティングリソースを共有しません。一部の Amazon GameLift Streams ストリームグループは、内部リソース共有に依存しています。

コンピューティングリソースの再利用

ストリームグループ内では、リソースは時間の経過とともに再利用され、最小限のダウントIMEで複数のセッションを提供します。再利用の具体的な詳細は、Windows ストリームグループと Windows 以外のストリームグループで異なります。

gen4n_high、、 gen6n_ultra などのストリームクラスを持つ Windows 以外のストリームグループは gen5n_ultra、セッションごとの専用コンテナ内でアプリケーション gen6n_pro を実行します。各ストリームセッションは、アプリケーションファイルのコピーと空のユーザー プロファイル フォルダで始まります。セッションが終了すると、すべてのファイルシステムの変更は破棄され、アプリケーションによって起動されたすべてのプロセスはコンテナクリーンアップの一部として終了します。

、 gen4n_win2022、、 などのストリームクラスを持つ Windows gen5n_win2022 ベースのストリームグループは gen6n_ultra_win2022、ホストオペレーティングシステムで直接アプリケーション gen6n_pro_win2022 を実行します。各ストリームセッションは、アプリケーションファイルのコピーと空のユーザー プロファイル フォルダで始まります。セッションが終了すると、ユーザー プロファイル フォルダとアプリケーション フォルダが完全にリセットされます。アプリケーションによって起動されたサブプロセスは終了します。アプリケーションがユーザー プロファイル フォルダとアプリケーション フォルダの外部でファイルを変更したり、システム レジストリを変更したりすると、それらの変更は複数のセッションにわたって保持される可能性があります。

ストリームグループ設定では、基盤となるコンピューティングリソースとオペレーティングシステム環境が時間の経過とともに再利用され、新しいストリームセッションが起動されます。[責任共有モデル](#)では、アプリケーションのセキュリティを維持し、信頼できないコードを実行したり、重要なオペレーティングシステムファイルを変更したりしないようにする責任があります。

マルチテナントストリームグループ

ストリームグループは、ストリームクラスの選択に応じて、シングルテナントまたはマルチテナントのいずれかです。gen4n_high やなどのマルチテナントストリームクラスは、複数の同時セッション間で 1 つの GPU gen5n_high を共有します。このコンテキストでは、マルチテナントとは、基盤となるハードウェアで一度に複数のセッションを実行することを意味します。ハードウェアは引き続きストリームグループ専用であり、ストリームグループ間や他の AWS お客様とは共有されません。

このマルチテナントストリームグループモデルは Amazon GameLift Streams に固有であり、セキュリティとパフォーマンスに重要な影響を与えます。マルチテナントストリームグループのセキュリティ体制は、単一の物理サーバーで複数のアプリケーションコンテナをホストすることと同じです。この体制は本質的に安全ではありませんが、アプリケーションにおける既存のセキュリティ脆弱性の影響を増幅する可能性があります。[責任共有モデル](#)では、アプリケーションのセキュリティを維持するのはお客様の責任です。

Amazon GameLift Streams は、マルチテナントセッションが相互に干渉しないように努めています。ただし、アプリケーションがストリームクラスの定義された制限に関係なく CPU または GPU リソースを消費すると、同じ共有リソースを使用しようとしている他のストリームに影響を与える可能性があります。たとえば、GPU ごとに 2 つのテナントを持つ「高」ストリームグループでは、greedy アプリケーションが最大 1 つの他のストリームに悪影響を及ぼす可能性があります。アプリケーションは独自のリソース消費を規制する必要があります。アプリケーションが自己制御できず、ユースケースに潜在的な「ノイズの多い隣接」パフォーマンスの変動に対する耐性がない場合は、gen5n_win2022、gen6n_pro_win2022、gen5n_ultraなどのシングルテナントストリームクラスgen6n_ultraをお勧めします。

Amazon GameLift Streams のインターフェイス VPC エンドポイント

インターフェイス VPC エンドポイントを使用するように Amazon GameLift Streams を設定することで、VPC のセキュリティ体制を改善できます。インターフェイスエンドポイントは AWS PrivateLink、プライベート IP アドレスを使用して Amazon GameLift Streams APIs にプライベートにアクセスできるテクノロジーであるを利用しています。AWS PrivateLink は、VPC と Amazon GameLift Streams 間のすべてのネットワークトラフィックを Amazon ネットワークに制限します。

インターネットゲートウェイ、NAT デバイス、または仮想プライベートゲートウェイは必要ありません。

AWS PrivateLink および VPC エンドポイントの詳細については、「Amazon [VPC ユーザーガイド](#)」の「[VPC エンドポイント](#)」を参照してください。

 Note

AWS PrivateLink は API エンドポイントにのみ適用されます。Amazon GameLift Streams マネージドストリームセッションは常にパブリックネットワークアドレスを使用します。

Amazon GameLift Streams の VPC エンドポイントの作成

Amazon GameLift Streams サービスの VPC エンドポイントを作成するには、「Amazon [VPC ユーザーガイド](#)」の「[インターフェイス VPC エンドポイントを使用して AWS サービスにアクセスする](#)」の手順を使用して、次のエンドポイントを作成します。

- com.amazonaws.*region*.gameliftstreams

 Note

region は、米国東部(オハイオ)リージョンなど、Amazon GameLift Streams で AWS リージョンサポートされている us-east-2 のリージョン識別子を表します。

Amazon GameLift Streams の VPC エンドポイントポリシーの作成

Amazon GameLift Streams へのアクセスを制御するエンドポイントポリシーを VPC エンドポイントにアタッチできます。このポリシーでは、以下の情報を指定します。

- アクションを実行できるプリンシパル。
- 実行可能なアクション。
- アクションを実行できるリソース。

詳細については、「Amazon VPC ユーザーガイド」の「[エンドポイントポリシーを使用して VPC エンドポイントへのアクセスを制御する](#)」を参照してください。

Example 例: Amazon GameLift Streams の VPC エンドポイントポリシー

以下は、Amazon GameLift Streams のエンドポイントポリシーの例です。エンドポイントにアタッチすると、このポリシーはストリームグループを作成して一覧表示するアクセス許可を付与します。

```
{  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Principal": "*",  
            "Action": [  
                "gameliftstreams>CreateStreamGroup",  
                "gameliftstreams>ListStreamGroups"  
            ],  
            "Resource": [  
                "*"  
            ]  
        }  
    ]  
}
```

Amazon GameLift Streams の設定と脆弱性の分析

設定と IT コントロールは、AWS お客様と当社のお客様との間の責任共有です。詳細については、AWS [「責任共有モデル」](#) を参照してください。AWS は、ゲストオペレーティングシステム (OS) やデータベースのパッチ適用、ファイアウォール設定、ディザスタリカバリなどの基本的なセキュリティタスクを処理します。これらの手順は適切なサードパーティによって確認され、認証されています。詳細については、[「Amazon Web Services: セキュリティプロセスの概要」](#) (ホワイトペーパー) を参照してください。

次のセキュリティのベストプラクティスは、Amazon GameLift Streams の設定と脆弱性の分析にも対処します。

- お客様は、ストリームホスティングのために Amazon GameLift Streams ストリームグループにデプロイされたソフトウェアの管理に責任を負います。具体的には次のとおりです。
 - 更新やセキュリティパッチなど、お客様が用意したアプリケーションコンテンツとソフトウェアを維持する必要があります。更新するには、新しい Amazon GameLift Streams アプリケーションを作成し、新しいストリームグループにデプロイします。

- 現時点では、ストリームグループのオペレーティングシステムとランタイム環境は、新しいストリームグループを作成する場合にのみ更新されます。ランタイム環境の一部であるオペレーティングシステムやその他のアプリケーションにパッチを適用、更新、保護するには、アプリケーションの更新に関係なく、2~4週間ごとにストリームグループをリサイクルすることをお勧めします。
- お客様は、SDK や Amazon GameLift Streams Web Client SDK などの最新の AWS SDK バージョンでゲームを定期的に更新することを検討する必要があります。

Amazon GameLift Streams のセキュリティのベストプラクティス

Amazon GameLift Streams には、独自のセキュリティポリシーを開発および実装する際に考慮すべきセキュリティ機能が多数用意されています。以下のベストプラクティスは一般的なガイドラインであり、完全なセキュリティソリューションを提供するものではありません。これらのベストプラクティスはお客様の環境に必ずしも適切または十分でない可能性があるため、処方箋ではなく、あくまで有用な検討事項とお考えください。

- 現時点では、ストリームグループのオペレーティングシステムとランタイム環境は、新しいストリームグループを作成する場合にのみ更新されます。ランタイム環境の一部であるオペレーティングシステムやその他のアプリケーションにパッチを適用、更新、保護するには、アプリケーションの更新に関係なく、2~4週間ごとにストリームグループをリサイクルすることをお勧めします。
- セキュリティ、アイデンティティ、コンプライアンスに関するベストプラクティス

Amazon GameLift ストリームのモニタリング

モニタリングは、Amazon GameLift Streams およびその他の AWS ソリューションの信頼性、可用性、パフォーマンスを維持する上で重要な部分です。AWS には、Amazon GameLift Streams をモニタリングし、問題が発生したときに報告し、必要に応じて自動アクションを実行するための以下のモニタリングツールが用意されています。

- Amazon CloudWatch は、AWS リソースとで実行しているアプリケーションを AWS リアルタイムでモニタリングします。メトリクスを収集および追跡し、カスタマイズされたダッシュボードを作成し、指定されたメトリックが指定したしきい値に達したときに通知またはアクションを実行するアラームを設定できます。詳細については、『[Amazon CloudWatch ユーザーガイド](#)』を参照してください。
- Amazon CloudWatch Logs を使用すると、Amazon Elastic Compute Cloud などのサービスやその他のソースからログファイルをモニタリング、保存 AWS CloudTrail、アクセスできます。CloudWatch Logs は、ログファイル内の情報をモニタリングし、サービスが特定のしきい値に達したときに通知できます。高い耐久性を備えたストレージにログデータをアーカイブすることもできます。詳細については、『[Amazon CloudWatch Logs ユーザーガイド](#)』を参照してください。
- AWS CloudTrail は、AWS アカウントによって、またはアカウントに代わって行われた API コールおよび関連イベントをキャプチャし、指定した Amazon Simple Storage Service バケットにログファイルを配信します。呼び出し元のユーザーとアカウント AWS、呼び出し元の送信元 IP アドレス、呼び出しの発生日時を特定できます。詳細については、『[AWS CloudTrail ユーザーガイド](#)』を参照してください。
- リアルタイムパフォーマンス統計は、ストリームセッション中にアプリケーションレベルと共有システムレベルのパフォーマンス統計を収集します。これらの統計は、クライアントでリアルタイムで受信することも、セッション後にエクスポートされたセッションファイルの CSV ファイルとして受信することもできます。この機能を使用すると、ストリームの CPU、メモリ、GPU、および VRAM 使用率をモニタリングできます。詳細については、『[the section called “リアルタイムのパフォーマンス統計”](#)』を参照してください。

Amazon CloudWatch で Amazon GameLift ストリームをモニタリングする

CloudWatch を使用して Amazon GameLift Streams をモニタリングできます。CloudWatch は raw データを収集し、読み取り可能なほぼリアルタイムのメトリクスに加工します。CloudWatch これら

の統計は 15 か月間保持されるため、履歴情報にアクセスし、ウェブアプリケーションまたはサービスの動作をより的確に把握できます。また、特定のしきい値を監視するアラームを設定し、これらのしきい値に達したときに通知を送信したりアクションを実行したりできます。詳細については、「[Amazon CloudWatch ユーザーガイド](#)」を参照してください。

Amazon GameLift Streams は、お客様が以下をモニタリングするのに役立つメトリクスを提供します。

- ・ストリームグループの容量と使用状況。
- ・ストリームパフォーマンスとリソース使用量。
- ・ストリームステータスを使用して問題を解決し、ユーザーをサポートします。
- ・コンテンツ提供全体のカスタマーエンゲージメント。
- ・データチャネルの使用状況。

次の表に、Amazon GameLift Streams のディメンションとメトリクスを示します。

ストリームグループの容量と使用状況

これらのメトリクスを使用して、需要に合わせてリソースをスケーリングします。これらのメトリクスは 1 分ごとに発行されます。

⚠ Important

2025 年 9 月 5 日より前に作成されたストリームグループの場合

CloudWatch のデータ保持ポリシーの問題により、正確な容量メトリクスは過去 15 日間のみ使用できます。15 日以上経過したキャパシティメトリクスの場合、期間が 1 分の場合、データは表示されず、期間が 5 分以上の場合、表示されるデータは不正確になります。

回避策として、CloudWatch グラフの sum-type 統計に `SUM(METRICS())/5` (5 分間の使用など) を回避策として追加して、15 日間の 1 分間のメトリクス保持制限を超える正確な容量数を確認できます。

この問題を修正するには、ストリームグループを再作成します。

メトリクス	説明	ディメンション	Unit
ActiveCapacity	プロビジョニングされ、ストリーミングする準備ができるコンピューティングリソースの数。これには、現在ストリーミング中のリソースと、新しいストリームリクエストに応答できるアイドル状態のリソースが含まれます。	(StreamGroupId、ロケーション)	カウント
IdleCapacity	現在ストリーミングされていないアクティブな容量の数値部分。これは、新しいストリームリクエストに応答するためのコンピューティングリソースの可用性を表します。	(StreamGroupId、ロケーション)	カウント

ストリームグループのパフォーマンスとリソース使用率

これらのメトリクスは 1 分ごとに発行されます。

メトリクス	説明	ディメンション	Unit
MemoryUtilization	ストリームで使用される使用可能なメモリの %。	(StreamGroupId、Location)、(ApplicationId、StreamClass)	パーセンテージ
CPUUtilization	ストリームで使用される使用可能な CPU の %。	(StreamGroupId、Location)、	パーセンテージ

メトリクス	説明	ディメンション	Unit
		(ApplicationId、StreamClass)	
FrameCaptureRate	アプリケーションからフレームがキャプチャされるレート。	(StreamGroupId、Location)、(ApplicationId、StreamClass)	なし
AudioCaptureRate	アプリケーションからオーディオサンプルがキャプチャされるレート。	(StreamGroupId、Location)、(ApplicationId、StreamClass)	なし
[RoundTripTime]	クライアントとサーバー間の往復時間。	(StreamGroupId、Location)、(ApplicationId、StreamClass)	ms

ストリームステータス

これらのメトリクスは、ストリームセッションの最後に発行されます。

メトリクス	説明	ディメンション	Unit
TerminatedStreamSessions	状態で終了したセッションの数 TERMINATED	(StreamGroupId、Location)	カウント

メトリクス	説明	ディメンション	Unit
		ation)、 (ApplicationId、 StreamClass)	
ErroredStreamSessions	状態で終了したセッションの数 ERROR	(StreamGroupId、 Location)、 (ApplicationId、 StreamClass)	カウント

カスタマーエンゲージメント

これらのメトリクスは、ストリームセッションの最後に発行されます。

メトリクス	説明	ディメンション	Unit
セッションの長さ	ストリームセッション期間	(StreamGroupId、 Location)、 (ApplicationId、 StreamClass)	秒

データチャネル

これらのメトリクスは、ストリームセッションの最後に発行されます。

メトリクス	説明	ディメンション	Unit
DataChannel-ApplicationConnected	アプリケーションがデータチャネルポートに接続する回数。この数は、ストリームセッションごとに最大 1 です。	(StreamGroupId、Location)、(ApplicationId、StreamClass)	カウント
DataChannel-ApplicationMessage	アプリケーションがクライアントに送信したメッセージの数。	(StreamGroupId、Location)、(ApplicationId、StreamClass)	カウント
DataChannel-ApplicationMessageBytes	アプリケーションがクライアントに送信したメッセージの合計バイト数。	(StreamGroupId、Location)、(ApplicationId、StreamClass)	バイト
DataChannel-ClientMessage	クライアントがアプリケーションに送信したメッセージの数。	(StreamGroupId、Location)、(ApplicationId、StreamClass)	カウント
DataChannel-ClientMessageBytes	クライアントがアプリケーションに送信したメッセージの合計バイト数。	(StreamGroupId、Location)、(ApplicationId、StreamClass)	バイト

を使用した Amazon GameLift Streams API コールのログ記録 AWS CloudTrail

Amazon GameLift Streams は、ユーザー[AWS CloudTrail](#)、ロール、またはによって実行されたアクションを記録するサービスであると統合されています AWS のサービス。CloudTrail は、Amazon GameLift Streams のすべての API コールをイベントとしてキャプチャします。キャプチャされた呼び出しには、Amazon GameLift Streams コンソールからの呼び出しと、Amazon GameLift Streams API オペレーションへのコード呼び出しが含まれます。CloudTrail で収集された情報を使用して、Amazon GameLift Streams に対するリクエスト、リクエスト元の IP アドレス、リクエスト日時などの詳細を確認できます。

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます。

- ルートユーザーまたはユーザー認証情報のどちらを使用してリクエストが送信されたか。
- リクエストが IAM Identity Center ユーザーに代わって行われたかどうか。
- リクエストがロールまたはフェデレーションユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。
- リクエストが、別の AWS のサービスによって送信されたかどうか。

CloudTrail は、アカウントを作成する AWS アカウントとでアクティブになり、CloudTrail イベント履歴に自動的にアクセスできます。CloudTrail の [イベント履歴] では、AWS リージョンで過去 90 日間に記録された管理イベントの表示、検索、およびダウンロードが可能で、変更不可能な記録を確認できます。詳細については、「AWS CloudTrail ユーザーガイド」の「[CloudTrail イベント履歴の使用](#)」を参照してください。[イベント履歴] の閲覧には CloudTrail の料金はかかりません。

AWS アカウント 過去 90 日間のイベントの継続的な記録については、証跡または[CloudTrail Lake](#) イベントデータストアを作成します。

CloudTrail 証跡

証跡により、CloudTrail はログファイルを Amazon S3 バケットに配信できます。を使用して作成されたすべての証跡 AWS マネジメントコンソールはマルチリージョンです。AWS CLIを使用する際は、単一リージョンまたは複数リージョンの証跡を作成できます。アカウント AWS リージョン内のすべてのアクティビティをキャプチャするため、マルチリージョン証跡を作成することをお勧めします。単一リージョンの証跡を作成する場合、証跡の AWS リージョンに記録

されたイベントのみを表示できます。証跡の詳細については、「AWS CloudTrail ユーザーガイド」の「[AWS アカウントの証跡の作成](#)」および「[組織の証跡の作成](#)」を参照してください。

証跡を作成すると、進行中の管理イベントのコピーを 1 つ無料で CloudTrail から Amazon S3 バケットに配信できますが、Amazon S3 ストレージには料金がかかります。CloudTrail の料金の詳細については、「[AWS CloudTrail の料金](#)」を参照してください。Amazon S3 の料金に関する詳細については、「[Amazon S3 の料金](#)」を参照してください。

CloudTrail Lake イベントデータストア

[CloudTrail Lake] を使用すると、イベントに対して SQL ベースのクエリを実行できます。CloudTrail Lake は、行ベースの JSON 形式の既存のイベントを [Apache ORC](#) 形式に変換します。ORC は、データを高速に取得するために最適化された単票ストレージ形式です。イベントは、イベントデータストアに集約されます。イベントデータストアは、[高度なイベントセレクタ](#)を適用することによって選択する条件に基づいた、イベントのイミュータブルなコレクションです。どのイベントが存続し、クエリに使用できるかは、イベントデータストアに適用するセレクタが制御します。CloudTrail Lake の詳細については、AWS CloudTrail ユーザーガイドの[AWS CloudTrail 「Lake の使用」](#)を参照してください。

CloudTrail Lake のイベントデータストアとクエリにはコストがかかります。イベントデータストアを作成する際に、イベントデータストアに使用する[料金オプション](#)を選択します。料金オプションによって、イベントの取り込みと保存にかかる料金、および、そのイベントデータストアのデフォルトと最長の保持期間が決まります。CloudTrail の料金の詳細については、「[AWS CloudTrail の料金](#)」を参照してください。

CloudTrail で Amazon GameLift Streams データイベント

[データイベント](#)は、リソースで、またはリソースで実行されるリソースオペレーションに関する情報を提供します(ストリームグループでストリームセッションを開始するなど)。これらのイベントは、データプレーンオペレーションとも呼ばれます。データイベントは、多くの場合、高ボリュームのアクティビティです。デフォルトでは、CloudTrail はデータイベントをログ記録しません。CloudTrail [イベント履歴] にはデータイベントは記録されません。

追加の変更がイベントデータに適用されます。CloudTrail の料金の詳細については、「[AWS CloudTrail の料金](#)」を参照してください。

CloudTrail コンソール、または CloudTrail CloudTrail API オペレーションを使用して AWS CLI、Amazon GameLift Streams リソースタイプのデータイベントを記録できます。データイベントをログに記録する方法の詳細については、「AWS CloudTrail ユーザーガイド」の「[AWS マネジメン](#)

[トコンソールを使用したデータイベントのログ記録](#)」および「[AWS Command Line Interfaceを使用したデータイベントのログ記録](#)」を参照してください。

次の表に、データイベントを記録できる Amazon GameLift Streams リソースタイプを示します。リソースタイプ(コンソール)列には、CloudTrail コンソールの [リソースタイプ] リストから選択する値が表示されます。resources.type 値列には、AWS CLI または CloudTrail APIs を使用して高度なイベントセレクタを設定するときに指定する resources.type 値が表示されます。CloudTrail に記録されたデータ API 列には、リソース タイプの CloudTrail にログ記録された API コールが表示されます。

リソースタイプ(コンソール)	resources.type 値	CloudTrail にログ記録されたデータ API
GameLift Streams アプリケーション	AWS::GameLiftStreams::Application	<ul style="list-style-type: none"> StartStreamSession
GameLift Streams ストリーム グループ	AWS::GameLiftStreams::StreamGroup	<ul style="list-style-type: none"> CreateStreamSessionConnection ExportStreamSessionFiles GetStreamSession ListStreamSessions ListStreamSessionsByAccount StartStreamSession TerminateStreamSession

eventName、readOnly、および resources.ARN フィールドでフィルタリングして、自分にとって重要なイベントのみをログに記録するように高度なイベントセレクタを設定できます。オブジェクトの詳細については、「AWS CloudTrail API リファレンス」の「[AdvancedFieldSelector](#)」を参照してください。

CloudTrail での Amazon GameLift Streams 管理イベント

[管理イベント](#)は、のリソースで実行される管理オペレーションに関する情報を提供します AWS アカウント。これらのイベントは、コントロールプレーンオペレーションとも呼ばれます。CloudTrail は、デフォルトで管理イベントをログ記録します。

Amazon GameLift Streams は、次の Amazon GameLift Streams コントロールプレーンオペレーションを管理イベントとして CloudTrail に記録します。

- [AddStreamGroupLocations](#)
- [AssociateApplications](#)
- [CreateApplication](#)
- [CreateStreamGroup](#)
- [DeleteApplication](#)
- [DeleteStreamGroup](#)
- [DisassociateApplications](#)
- [GetApplication](#)
- [GetStreamGroup](#)
- [ListApplications](#)
- [ListStreamGroups](#)
- [ListTagsForResource](#)
- [RemoveStreamGroupLocations](#)
- [TagResource](#)
- [UntagResource](#)
- [UpdateApplication](#)
- [UpdateStreamGroup](#)

Amazon GameLift Streams イベントの例

各イベントは任意の送信元からの単一のリクエストを表し、リクエストされた API オペレーション、オペレーションの日時、リクエストパラメータなどに関する情報を含みます。CloudTrail ログファイルは、パブリック API コールの順序付けられたスタックトレースではないため、イベントは特定の順序で表示されません。

次の例は、CreateApplication オペレーションを示す CloudTrail 管理イベントを示しています。

[CreateApplication](#)

```
{  
    "eventVersion": "1.09",  
    "userIdentity": {  
        "type": "AssumedRole",  
        "principal": "arn:aws:iam::123456789012:role/GameLift-GameServer-Role",  
        "accountId": "123456789012",  
        "accessKeyId": "ASIAQWERTYUICL",  
        "sessionContext": {  
            "sessionIssuer": {  
                "arn": "arn:aws:sts::123456789012:assumed-role/GameLift-GameServer-Role/123456789012",  
                "type": "AWSIdentityProvider"  
            },  
            "contextParameters": "",  
            "creationDate": "2023-01-12T12:00:00Z",  
            "expirationDate": "2023-01-12T12:05:00Z",  
            "lastUsedDate": "2023-01-12T12:00:00Z",  
            "arn": "arn:aws:sts::123456789012:assumed-role/GameLift-GameServer-Role/123456789012",  
            "principal": "arn:aws:iam::123456789012:role/GameLift-GameServer-Role",  
            "contextConditions": ""  
        }  
    }  
}
```

```
"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"
        },
        "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-EXAMPLE1111",
"eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbbb",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333",
"eventCategory": "Management"
}
```

次の例は、[StartStreamSession](#) オペレーションを示す証跡ログからの CloudTrail データイベントを示しています。

```
{
    "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"
                    },
                    "contextParameters": {}
                }
            }
        }
    ]
}
```

```
        "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"  
}  
]  
}
```

CloudTrail レコードの内容については、「AWS CloudTrail ユーザーガイド」の「[CloudTrail record contents](#)」を参照してください。

リアルタイムのパフォーマンス統計

Amazon GameLift Streams は、アクティブなストリームセッション中にパフォーマンス統計を収集し、毎秒リソース使用率を測定します。これらの統計を使用して、アプリケーションのパフォーマンスをモニタリングし、リソースのボトルネックを特定し、ストリーミングエクスペリエンスを最適化します。

パフォーマンス統計には、アプリケーションレベルの統計 (特定のアプリケーションの CPU とメモリの使用率) とシステムレベルの統計 (共有コンピューティングインフラストラクチャの CPU、メモリ、GPU、VRAM の使用率) の両方が含まれます。

パフォーマンス統計は、次の 2 つの方法で受け取ることができます。

- セッション中にリアルタイムで: Amazon GameLift Streams Web SDK を使用して、収集された統計情報を受け取ります。これにより、パフォーマンスオーバーレイを構築し、アプリケーションを操作する際のリソース使用率をモニタリングできます。
- CSV ファイルとしてのセッション後: セッションファイルをエクスポートすると、統計はとして含まれます `stats/perf_stats_v1.csv`。これにより、セッション後の分析とデバッグの完全なレコードが提供されます。

パフォーマンス統計を受信する

統計をリアルタイムで受信する

アクティブなセッション中にクライアントアプリケーションでパフォーマンス統計を受信するには、`StartStreamSession` API を呼び出す `true` ときに `SharedWithClient` パラメータをに設定します。Amazon GameLift Streams Web SDK は、ストリーミングセッションから新しい統計が到着するたびにトリガーする `performanceStats` コールバックを提供します。

Warning

エンドユーザーとの本稼働セッション `SharedWithClient` ではを有効にしないでください。内部デバッグやテストなど、クライアントが信頼されている場合にのみ有効にします。

Amazon GameLift Streams Web SDK を初期化するときは、パフォーマンス統計を受信するコールバック関数 `clientConnection.performanceStats` に設定します。

```
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 + '%');
        },
    }
});
```

コールバックは、アプリケーションレベルとシステムレベルの両方の統計を含むPerformanceStatsオブジェクトを受け取ります。インターフェイス構造の詳細については、[入門製品ページの Amazon GameLift Streams Web SDK ドキュメント](#)を参照してください。

Amazon GameLift Streams コンソールには、テストストリーム機能を使用する際のパフォーマンスオーバーレイも組み込まれているため、実装作業を行わずにリアルタイムで統計をモニタリングできます。

Amazon GameLift Streams Web SDK の および 関数によって提供される WebRTC 統計とパフォーマンス統計を組み合わせることができます。getVideoRTCStats() getAudioRTCStats() この組み合わせは、ネットワーク統計、クライアントフレームレート、リソース使用率など、ストリーミングパフォーマンスの全体像を提供します。

セッション後の統計を受信する

Amazon GameLift Streams は、すべてのストリームセッション中にパフォーマンス統計を自動的に収集します。セッションファイルをエクスポートすると、統計はエクスポートされた ZIP ファイルstats/perf_stats_v1.csv内のとして含まれます。これにより、セッション後の分析とデバッグのためにセッション中に収集されたすべての統計の完全な記録が提供されます。

セッションファイルのエクスポートの詳細については、「」を参照してください[the section called “ストリームセッションファイルのエクスポート”。](#)

パフォーマンス統計リファレンス

次の表に、Amazon GameLift Streams によって収集されたすべてのパフォーマンス統計を示します。アプリケーション統計は現在のセッションに固有ですが、共有システム統計はマルチテナントストリームクラスのセッションごとの共有コンピューティングの合計使用率を反映しています。

i マルチテナントストリームクラスの正規化された統計

Amazon GameLift Streams は、複数のセッションが同じコンピューティングインスタンスを共有するマルチテナントストリームクラスをサポートしています。正規化された統計(アプリケーションの CPU とメモリの使用率)は、割り当てられた公平配分に対するアプリケーションのリソース使用率を測定します。公平配分は、ストリームクラスのテナンシーに基づいてコンピューティングインスタンスで使用可能な CPU とメモリの合計を均等に割って計算されます。

値 1.0 は、アプリケーションが公平配分を正確に使用していることを意味します。1.0 未満の値は、が割り当てより少ないことを示します。値が 1.0 を超えると、使用率が高くなり、セッションのパフォーマンスが低下する可能性があります。マルチテナントストリームクラス(テナンシーが 1 より大きい)では、使用率が高いと、同じコンピューティングインスタンスを共有する他のセッションにも影響する可能性があります。

次の表に示す統計名は、エクスポートされたファイルの CSV 列ヘッダーとして使用されます。Amazon GameLift Streams Web SDK を介してリアルタイムで統計を受け取る場合、これらの統計はキャメルケースのプロパティ名を持つ PerformanceStats インターフェイスを介して使用できます。正確なインターフェイス構造とプロパティ名については、入門製品ページの「Amazon GameLift Streams Web SDK API リファレンスガイド」を参照してください。 [https://aws.amazon.com/gamelift\(streams/getting-started/](https://aws.amazon.com/gamelift(streams/getting-started/)

統計名(CSV 列)	説明	[Scope] (スコープ)
timestamp	ISO 8601 形式で測定が行われた時刻。	すべて
app_cpu_normalized	公平配分に対して正規化されたアプリケーションの CPU 使用率。1.0 は目標公平配分制限を表します。1.0 を超える使用量は過剰使用率を示し、パフォーマンスの問題につながる可能性がある	アプリケーション

統計名 (CSV 列)	説明	[Scope] (スコープ)
app_mem_mb	アプリケーションで使用される合計メモリ (RAM) (MiB で測定)	アプリケーション
app_mem_normalized	公平配分割合に対して正規化されたアプリケーションのメモリ使用量。1.0 は目標公平配分制限を表します。1.0 を超える使用量は過剰使用率を示し、パフォーマンスの問題につながる可能性がある	アプリケーション
shared_systems_cpu_pct	共有コンピューティング全体の合計 CPU 使用率。	共有システム
shared_systems_mem_mb	インスタンスで使用されている合計メモリ (MiB で測定)。	共有システム
shared_systems_mem_pct	共有コンピューティング全体で使用されている合計メモリの割合。	共有システム
shared_systems_gpu_pct	共有コンピューティング全体の GPU 使用率の合計の割合。	共有システム
shared_systems_vram_mb	共有コンピューティングで使用される合計 VRAM (GPU メモリ) (MiB で測定)。	共有システム
shared_systems_vram_pct	共有コンピューティング全体で使用されている合計 VRAM (GPU メモリ) の割合。	共有システム

Amazon GameLift Streams のトラブルシューティング

トピック

- [Amazon GameLift Streams サービスへのリクエスト時にアクセスが拒否されました](#)
- [アプリケーションに関する問題](#)
- [パフォーマンスの問題](#)
- [ストリーム接続とネットワークパフォーマンスの問題](#)
- [ストリーム入力の問題](#)
- [ストリームセッションの問題](#)
- [Proton for Amazon GameLift Streams との互換性のテストとトラブルシューティング](#)
- [Unreal Engine のパフォーマンスのプロファイリング](#)

Amazon GameLift Streams サービスへのリクエスト時にアクセスが拒否されました

Amazon GameLift Streams APIs を呼び出すとき、またはコンソールでリソースを操作するときに AccessDenied 例外が発生した場合、AWS Identity and Access Management (IAM) ロールに Amazon GameLift Streams に対する十分なアクセス許可がない可能性があります。以下をチェックしてください:

- IAM ロールに明示的な「すべて拒否」ポリシーがある場合は、[NotAction](#) 要素に を追加して、Amazon GameLift Streams "gameliftstreams:*"をそのポリシーの例外として明示的にリストする必要があります。例えば、次のようになります。

```
{  
    "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"}  
    }  
}
```

- 詳細については、「セキュリティ」章[Amazon GameLift Streams の Identity and Access Management](#)の「」および「IAM ユーザーガイド」の[「アクセス拒否エラーメッセージのトラブルシューティング」](#)を参照してください。

アプリケーションに関する問題

このセクションでは、アプリケーションの実行を妨げたり、Amazon GameLift Streams で表示が異なっていたりする問題の潜在的な原因を特定します。

事前チェック

- 別のマシンでアプリケーションを実行して、正しくパッケージ化されていることを確認します。これにより、アプリケーションコンテンツに、他のデバイスで機能しない可能性のあるハードコードされたパス、欠落しているアセット、ライブラリ、またはバイナリが含まれていないことを確認します。

Proton の問題

- アプリケーションが Proton と互換性があることを確認します。Amazon GameLift Streams サーバーを使用せずにローカル環境でアプリケーションをテストし、Proton と互換性があることを確

認めます。手順については、「[Proton for Amazon GameLift Streams との互換性のテストとトラブルシューティング](#)」を参照してください。

画面解像度によるアプリケーションの問題

1920x1080 ではない全画面解像度を使用しようとすると、アプリケーションがフリーズ、クラッシュ、または正しくレンダリングされない可能性があります。ボーダーレス全画面ウィンドウを使用してアプリケーションを実行し、解像度を変更しないことをお勧めします。

ストリームセッションの開始時にアプリケーションが終了する

ストリームセッションの開始時にアプリケーションがすぐに終了する場合は、考えられる原因と解決策について以下を確認してください。

- ・ ランタイムを確認します。Amazon GameLift Streams アプリケーション設定で、実行可能起動パスで指定したファイルが実行可能ファイルまたはスクリプトであり、選択したランタイム環境に適していることを確認します。Windows アプリケーションは、ファイルタイプが「.exe」、「.bat」、または「.cmd」で、Windows または Proton ランタイムのいずれかをターゲットにする必要があります。ネイティブ Linux アプリケーションは、Ubuntu 22.04 LTS ランタイムをターゲットとする実行可能ファイルである必要があります。
- ・ 必要な DLLs を確認します。Windows アプリケーションに必要な DLLsがない可能性があります。たとえば、アプリケーションがデバッグビルドの場合、Microsoft C および C++ (MSVC) ランタイムライブラリのデバッグバージョンが必要です。これを解決するには、ビルドと DLLs side-by-side/パッケージ化することをお勧めします。手順については、「[Microsoft によるデバッグ実行可能なファイルを実行するテストマシンを準備する](#)」を参照してください。

一般的に、Amazon GameLift Streams を試す前に、最初にクリーンマシンでビルドをテストすることをお勧めします。Amazon EC2 インスタンスでのテストの手順については、「」を参照してください。[リモートマシンをセットアップする](#)。

Unreal Engine アプリケーションがクラッシュするか、追加の依存関係が必要です

Unreal Engine アプリケーションの起動に失敗したり、クラッシュしたり、Microsoft C や C++ (MSVC) ランタイムなどの追加の依存関係をインストールする必要がある場合は、以下を試してください。

- 正しい実行可能ファイルを使用します。アプリケーションが Amazon GameLift Streams で正しく動作するには、アプリケーションパスを Binaries/Win64/ (または同様の) サブフォルダにある完全な実行可能ファイルに設定します。Unreal Engine は 2 つの実行可能ファイルを生成します。1 つはフォルダのルートに小さなブートストラップ実行可能ファイル、もう 1 つはBinaries/Win64/サブフォルダにプラットフォームターゲット実行可能ファイルです。前提条件を検証しようとするルート試行時のブートストラップ実行可能ファイルが正しいため、Amazon GameLift Streams でアプリケーションの起動を妨げる誤検出が発生する可能性があります。プラットフォームターゲット実行可能ファイルがない場合、アプリケーションが正しく構築されていない可能性があります。たとえば、サンプルの Unreal アプリケーションの次のフォルダ構造を参照してください。

```
BuildApp
| -> MyUnrealApp.exe
| -> MyUnrealApp
    | -> Binaries
        | -> Win64
            | -> MyUnrealApp.exe
```

- Unreal Engine Asserts をオフにします。check、verify、および ensure マクロを無効にします。これにより、アプリケーションがクラッシュダンプを作成できなくなる可能性があります。詳細については、[Unreal Engine ドキュメントの「アサート」](#)を参照してください。
 - ビルド USE_CHECKS_IN_SHIPPING=0 を定義して、check マacro と verify マacro を無効にします。
 - handleensurepercent=0 コマンドライン引数を使用して ensure マcro を無効にします。

パフォーマンスの問題

このセクションでは、Amazon GameLift Streams で実行する際のゲームパフォーマンスの問題の潜在的な原因を特定し、サービスでストリームを最適化するための提案を提供します。

Amazon GameLift Streams でストリーミングするとゲームのパフォーマンスが低下する

ゲームが独自のマシンでうまく動作しているが、Amazon GameLift Streams でストリーミングするときにパフォーマンスの問題が発生した場合は、次の点を考慮してください。

- マシンには、Amazon GameLift Streams よりも強力なハードウェアがある可能性があります。Amazon GameLift Streams が使用するハードウェアと同様のパフォーマンスのマシンでアプリケーションをテストしてください。
 - gen4n: NVIDIA RTX 2060 GPU と同等
 - gen5n: NVIDIA RTX 3080 GPU と同等
 - gen6n: NVIDIA RTX 4060 GPU と同等

これにより、アプリケーションのレンダリング設定が GPU と互換性があり、パフォーマンスが期待を満たしていることを確認します。

- 問題は、ネットワーク接続または Amazon GameLift Streams の設定が原因である可能性があります。[ストリーム接続の問題](#) 「」セクションのトラブルシューティングのヒントを試してください。

ローカルで実行してもゲームが遅い場合は、そのパフォーマンスを最適化する必要があります。最適な最適化方法は、使用している特定のエンジンまたはフレームワークによって異なります。

- Unreal Engine ゲームについては、「」を参照してください[Unreal Engine のパフォーマンスのプロファイリング](#)。

Windows アプリケーションでは、ロード時間が遅い、またはスタッタリングの問題が発生する

ゲームのロード時間が長いか、スタッタリング動作が発生している場合は、次の一連のアクションをお勧めします。

- コンテンツとシェーダーのパフォーマンスの最適化に関するエンジンベンダーのガイダンスを使用して、アプリケーションがパッケージ化され、ロードパフォーマンス用に最適化されていることを確認します。
- アプリケーションがストリームグループの[デフォルトアプリケーション](#)に設定されていることを確認します。
- アプリケーションパッケージの一部としてシェーダーをキャッシュすることで、サービスでのアプリケーションの初回起動を最適化します。

シェーダーキャッシュを有効にするには、次の 2 つの方法があります。

- ドライバーベースのキャッシュ – このアプローチは、ランタイム環境の GPU とドライバーのバージョンに固有のものです。このオプションはすべてのアプリケーションに適用できるため、デフォルトの推奨アプローチです。このアプローチのステップは、GPU/ドライバーの組み合わせごとにレプリケートする必要があります。
- エンジンベースのキャッシュ – このアプローチにより、利用可能な場合はゲームエンジンを介したシェーダーキャッシュが可能になります。これにより、事前にベイクされたパイプライン状態オブジェクト (PSO) キャッシュを開発者に作成する負担がかかります。また、エンジンが同じ GPU ハードウェア上の異なるドライバーのキャッシュサポートを処理できることを前提としています。

ベストプラクティスとして、まずドライバーベースのキャッシュを実装することをお勧めします。これは、特定のエンジンに PSO キャッシュがどのように実装されるかを深く理解する必要がないためです。

これらの実装では、シェーダーファイルをエクスポートしてアプリケーションにパッケージ化できるため、新しいストリームの開始ごとに生成する必要はありません。

Windows ランタイムアプリケーションのドライバーベースのキャッシュ修正を実装するには

1. デフォルトのアプリケーションのストリーミングを開始し、広範囲に再生してアプリケーションのシェーダーを生成します。

⚠ Important

可能な限り多くのシェーダーを生成するには、必ず環境のすべての領域またはレベルにアクセスしてください。

2. ストリームを閉じる前に、アクティブなストリームセッションでエクスポート機能を有効にします。詳細については、「[ストリームセッションファイルのエクスポート](#)」を参照してください。
3. 前のステップで指定した Amazon S3 バケットからストリームセッションエクスポート .zip ファイルをダウンロードします。ダウンロードリンクは、セッションページの Amazon GameLift Streams コンソールにあります。
4. ストリームセッションエクスポート内のシェーダーフォルダを見つけます。通常、という場所に保存されます AppData\Local\NVIDIA\DXCache。生成されたシェーダーファイル (*.nvph) をアプリケーションの Amazon S3 バケットにアップロードします。
5. 実行時にシェーダー. bat ファイルを NVIDIA キャッシュフォルダにコピーするファイルを作成します。このフォルダは通常、にあります C:\Users\Administrator\AppData\Local

- \NVIDIA\DXCache。.bat ファイルを Amazon S3 アプリケーションバケットにアップロードします。
6. ファイルを実行可能なパス.batとして新しい Amazon GameLift Streams アプリケーションを作成します。

アプリケーションがストリーミングを開始すると、.bat ファイルはアプリケーションを起動する前に事前に生成されたシェーダーをシェーダーキャッシュにコピーし、ストリームのロードパフォーマンスを向上させます。

 Note

アプリケーションを更新したり、Amazon GameLift Streams アプリケーションを新しいストリームグループにリンクしたりするたびに、これらのステップを繰り返す必要がある場合があります。新しいストリームグループには、サービスから更新された GPU ドライバーを含めることができます。

次のサンプル.bat ファイルは、シェーダーファイルが Amazon S3 バケットプレフィックスに保存されていることを前提としています Shaders\。別のフォルダ構造を使用できます。

```
@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
```

Proton ランタイムアプリケーションのドライバーベースのキャッシュ修正を実装するには

1. 次の環境変数オーバーライドを使用して、デフォルトのアプリケーションのストリーミングを開始します。

```
"__GL_SHADER_DISK_CACHE_PATH" : "/home/unpriv/games"
```

2. アプリケーションを広範囲に再生してシェーダーを生成します。

⚠ Important

可能な限り多くのシェーダーを生成するには、必ず環境のすべての領域またはレベルにアクセスしてください。

3. ストリームを閉じる前に、アクティブなストリームセッションでエクスポート機能を有効にします。詳細については、「[ストリームセッションファイルのエクスポート](#)」を参照してください。
4. 前のステップで指定した Amazon S3 バケットからストリームセッションエクスポート .zip ファイルをダウンロードします。ダウンロードリンクは、セッションページの Amazon GameLift Streams コンソールにあります。
5. ストリームセッションエクスポート内のシェーダーフォルダとファイルを見つけます。
 - a. application\GLCache フォルダ
 - b. アプリケーションで DX11 を使用している場合: application\path-to-exe\exe-name.dxvk-cache ファイル
 - c. アプリケーションで DX12 を使用している場合: application\path-to-exe\vkd3d-proton.cache.write ファイル
6. 生成されたシェーダーファイルをアプリケーションの Amazon S3 バケットにアップロードします。
 - a. GLCache フォルダをアプリケーションのルートディレクトリにコピーします。
 - b. 可能であれば、.dxvk-cache または vkd3d-proton.cache.write キャッシュファイルをアプリケーション実行可能ファイルを含むフォルダにコピーします。
7. 同じ Proton 設定で新しい Amazon GameLift Streams アプリケーションを作成します。
8. 同じ環境変数オーバーライドでアプリケーションを実行します。

```
"__GL_SHADER_DISK_CACHE_PATH" : "/home/unpriv/games"
```

アプリケーションがストリーミングを開始すると、事前に生成されたシェーダーが使用され、ストリームのロードパフォーマンスが向上します。

Note

アプリケーションを更新したり、Amazon GameLift Streams アプリケーションを新しいストリームグループにリンクしたりするたびに、これらのステップを繰り返す必要がある場合があります。新しいストリームグループには、サービスから更新された GPU ドライバーを含めることができます。

Unreal Engine を使用してアプリケーションのエンジンベースのキャッシュ修正を実装するには

このアプローチでは、Unreal Engine 機能を使用して、Amazon GameLift Streams アプリケーションのパイプライン状態オブジェクト (PSO) キャッシュを作成できます。PSO キャッシュを使用すると、事前コンパイルされたグラフィックスパイプラインの状態をランタイムコンパイル時間を短縮して配信できるため、ロードとレンダリング中のハイッチを減らすことができます。これには Unreal Engine に関する高度な知識が必要なため、ここではエンジン固有の詳細をすべて網羅するわけではありません。その他の手順については、[バンドルされた PSO キャッシュの作成](#)、「コレクションフォー」セクションの「Unreal Engine」のガイダンスを参照してください。

1. PSO ログ記録が有効になっているアプリケーションのシェーダーを生成します。
 - a. PSO 対応アプリケーションでパッケージ化されたビルドを使用して、新しい Amazon GameLift Streams アプリケーションを作成します。
 - b. PSO ログ記録アプリケーションで `-logPSO` コマンドを使用してストリームを開始します。Amazon GameLift Streams コンソールのストリーム設定のテストページで、コマンドライン引数オプションを使用できます。

Important

可能な限り多くのシェーダーを生成するには、必ず環境のすべての領域またはレベルにアクセスしてください。

- c. ストリームを閉じる前に、アクティブなストリームセッションでエクスポート機能を有効にします。詳細については、[「ストリームセッションファイルのエクスポート」](#)を参照してください。
- d. メニューまたは Unreal シャットダウンコマンドを使用して、アプリケーションを終了します。ストリームを直接閉じた場合、Unreal シェーダーバイオペラインファイルは生成されません。

- e. エクスポートステップで指定した Amazon S3 バケットからストリームセッションエクスポート .zip ファイルをダウンロードします。ダウンロードリンクは、セッションページの Amazon GameLift Streams コンソールにあります。
2. Unreal シェーダーパイプラインファイルを Amazon GameLift Streams アプリケーションにパッケージ化します。
 - a. のストリームセッションエクスポートで、記録された PSO ファイル (rec.pipelinecache) を見つけます Saved/CollectedPSOs。Unreal コマンドを使用して PSO ファイルを解凍します。
 - b. 解凍から生成された出力を使用して、新しい Unreal ビルドをパッケージ化します。Unreal ガイダンスのセクション [「PSO キャッシュの変換」](#) および [「アプリケーションに PSO キャッシュを含める」](#) に従ってください。

 **Important**

「PSO キャッシュの変換」セクションで Unreal コマンドを実行するときは、同じドライバーのバージョン入力ファイルを使用していることを確認してください。例: DX12 では、入力として SM6 ファイルのみを使用します。そうしないと、新しいアプリケーションのパッケージ化時にエラーが発生します。

- c. PSO ファイルを使用して、新しいパッケージ化されたビルド用の新しい Amazon GameLift Streams アプリケーションを作成します。
- d. ストリームを起動してテストするときは、PSO キャッシュがロードされていることを確認します。ゲームログで次の行を確認します。

Opened FPipelineCacheFile: ../../...

 **Note**

アプリケーションを更新したり、Amazon GameLift Streams アプリケーションを新しいストリームグループにリンクしたりするたびに、これらのステップを繰り返す必要がある場合があります。新しいストリームグループには、サービスから更新された GPU ドライバーを含めることができます。

ストリーム接続とネットワークパフォーマンスの問題

[Amazon GameLift Streams バックエンドサービスを設定する](#)ときは、以下を確認してください。

- エンドユーザー AWS リージョン に最も近いものを選択します。クライアントからストリームをホストするリージョンへのレイテンシーが高いと、ストリームの品質に影響する可能性があります。ストリーミング元の場所のリスト[AWS リージョン Amazon GameLift Streams でサポートされている およびリモートロケーション](#)については、「」を参照してください。リージョンの AWS コンソールエンドポイントに ping を実行して、およそのレイテンシー測定値を取得できます。
- ストリームグループに新しいストリームの容量があることを確認します。
- ウェブクライアントConnectionTimeoutSecondsがタイムアウトする前に、エンドユーザーが接続に十分な時間を確保できるようにが合理的に設定されていることを確認します。

エンドユーザーに以下を確認するようアドバイスします。

- ファイアウォールが UDP ポート範囲 33435-33465 へのアクセスを許可し、Amazon GameLift Streams からのストリーミングを許可していることを確認します。Amazon GameLift Streams がこれらのポートに到達できない場合、黒や灰色の画面などのストリーミングの問題が発生する可能性があります。
- インターネット接続が 1080p ストリームで少なくとも 10 Mbps の接続速度を維持できることを確認します。Amazon GameLift Streams で再生中にネットワークの問題を検出すると、インターネット速度が変動し、少なくとも 10 Mbps が一貫して得られない可能性があります。インターネット速度テストを実行し、トラブルシューティングの手順に進みます。
- 可能であれば、有線ネットワークを使用します。Wi-Fi を使用する場合は、デバイスをルーターの近くに移動して、信号強度を強化します。
- 2.4 GHz 帯と 5 GHz 帯の両方の Wi-Fi ルーターを使用している場合は、別の帯に接続してみてください。ルーターを別のバンドに切り替える方法がわからない場合は、Wi-Fi ルーターの製造元またはプロバイダーのサポートページにアクセスしてください。カスタマーサービスに連絡することもできます。
- 同じネットワーク上の他のユーザー (特にホーム Wi-Fi の場合) が、ビデオストリーミング、ダウンロード、オンラインゲーム、バックアップなどの高帯域幅アプリケーションを実行しているかどうかを特定します。
- 帯域幅を使用するデバイス上の他のアプリケーションを閉じます。
- ストリーミング中に VPN またはプロキシを使用しないでください。レイテンシーが高くなり、ゲームプレイに影響を与える可能性があります。

- iPad または iPhone で再生するときに、セルラーネットワークの代わりに Wi-Fi を使用していることを確認します。セルラーネットワークを使用すると、接続の問題が発生する可能性があります。
- MacOS ユーザーは、Wi-Fi が時々一時停止し、ストリーミングエクスペリエンスが低下するため、位置情報サービスを無効にする必要があります。

ストリーム入力の問題

このセクションでは、ストリームセッションでのユーザー入力に関する問題の潜在的な原因と解決策を特定します。

一般的な入力のトラブルシューティング

- テストして、問題が browser-specific かどうかを確認します。全体として、特にゲームコントローラーとの互換性を最大限に高めるために、Google Chrome、Microsoft Edge、またはカスタム Chromium ベースのデスクトップアプリケーションをお勧めします。
- クライアントから送信され、アプリケーションによって受信された入力イベントをログに記録し、フロントエンドコードの入力ミスマッチがある場所を特定します。
- 既知の問題や制限など、サポートされているブラウザと入力デバイスに関する [サポートされているブラウザと入力](#) 追加情報を必ず確認してください。

ゲームパッドとマイクの入力がネイティブ Linux アプリケーションで機能しない

ゲームパッドとマイクの入力は、ネイティブ Linux アプリケーションではサポートされていません。既知の問題や制限など、サポートされている入力デバイスの詳細については、[サポートされているブラウザと入力](#) 「」を参照してください。

MacOS クライアントでキー入力がスタックしているように見える

MacOS クライアントでは、コマンド修飾キーと別のキーを同時に押してキーイベントを繰り返すと、キーが突然停止しているように見えることがあります。たとえば、コマンドキーも押されると、矢印キーがスタックすることがあります。ゲームでは、矢印キーを使用してカメラを回転すると、カメラは無限に回転します。

- 問題: MacOS のコマンドキーは Meta キーイベントにマッピングされ、Microsoft Windows の Windows キーにマッピングされます。問題は、コマンドと別のキーを同時に押すと MacOS ブラ

ウザに影響するバグです。ここでは、リリース時に Meta キーがリセットされますが、ブラウザが矢印キーのキーアップイベントをキャプチャしなかったため、Web SDK クライアントはキーアップイベントをサーバーに送信せず、ストリーミングアプリケーションはキーが押されていると見なします。

- 解決策: コマンドキーを使用していない場合は、Web SDK の `InputConfiguration` インタフェイスにある Web SDK キーボードフィルターメカニズム (`keyboardFilter`) を使用してフィルタリングできます。

マウスの動きは Amazon GameLift Streams で動作が異なります

予想よりも速く移動するなど、Amazon GameLift Streams でストリーミングするときにマウスの動きの動作が異なる場合は、アプリケーションでマウス処理とカーソル管理のロジックを調整する必要がある場合があります。

- 問題: Amazon GameLift Streams はヒューリスティックを使用して、マウスイベントを「相対」モードで送信するか「絶対」モードで送信するかを選択します。相対モードでは、新しいマウス更新は、以前の更新との小さな増分的な違いとして提供されます。絶対モードでは、マウスカーソルはクライアントと同期された画面位置に継続的に強制されます。ストリーミングされたコンテンツにオペレーティングシステムカーソルが表示されると、ヒューリスティックは常に絶対座標を選択します。これにより、アプリケーションが小さな相対的な更新を想定している場合、予期しない大きな移動デルタが発生する可能性があります。
- 解決策: アプリケーションでマウスの相対モーション (FPS 形式のカメラコントロールやドラッグベースのインタラクションなど) が予想される場合は、マウスインタラクション中にオペレーティングシステムカーソルを非表示にします。たとえば、マウスダウンドラッグでカーソルを非表示にし、マウスアップで再度表示します。これにより、ドラッグモーションは相対座標を使用し、絶対位置はボタンが放されたときにのみ同期されます。

Amazon GameLift Streams でのマウスの動きの詳細については、「」を参照してください [マウスの移動処理](#)。

ストリームセッションの問題

このセクションでは、予期せず開始または終了するストリームセッションに関連する問題の潜在的な原因と解決策を特定します。

ストリームセッションが開始されない

考えられる原因:

- ・ アプリケーションがハングしているかクラッシュしています。トラブルシューティングの手順については、[アプリケーションに関する問題「」セクション](#)を参照してください。
- ・ ストリームグループのステータスは ではありません Active。ストリームグループのステータスを確認します。
- ・ オンデマンドキャパシティのスピンドアップには、[StartStreamSession API](#) ConnectionTimeoutSecondsで 指定されたタイムアウトよりも時間がかかります。Windows ランタイムでは、オンデマンドのスピンドアップに 5 分以上かかる場合があります。
- ・ ストリーミングロケーションに使用可能な容量がありません。割り当てられた容量が使用中の容量より大きいか、使用されていないオンデマンド容量があることを確認します (割り当てられた容量が常時オン容量にオンデマンド容量を加えたものより小さい)。コンソールでは、これらの値はストリームグループのリストまたはストリームグループの詳細ページで確認できます。サービス API を使用すると、[GetStreamGroup](#) を使用してこれらの値を見つけることができます。使用可能な容量が一時的にゼロになるシナリオには、次のようなものがあります。
 - ・ ストリーミングロケーションで常時オン容量を増やしたばかりの場合は、容量が割り当てられるまで数分待ちます。
 - ・ ストリーミングロケーションに使用可能な容量が 1 つしかなく、クライアントが予期せず切断された場合、前のセッションは切断された状態のままである可能性があります。セッションがタイムアウトするまで数分待ってから、もう一度試してください。
 - ・ 最近ストリームグループに場所を追加し、その場所にアプリケーションが存在しない場合、アプリケーションはそこでのレプリケーションを完了していない可能性があります。コンソールのストリームグループの詳細ページでレプリケーションステータスを確認します。または、[GetApplication API](#) を使用して ReplicationStatuses 値をチェックし、目的のストリーミング場所 Status のがであることを確認します COMPLETED。
 - ・ ネットワーク条件が非常に悪いため、フレーム、特に最初のフレームは送信されません。クライアントとストリーミングロケーション間のネットワーク状態を確認し、別のロケーションを調整または試します。

ストリームセッションが終了しました

ストリームセッションは、アプリケーションがクラッシュまたは終了したとき、またはクライアント接続が失われたときに自動的に終了します。セッションは、次のタイムアウト値が原因で終了することもできます。

- 配置タイムアウト: ストリームセッションをホストするコンピューティングリソースを見つけるための Amazon GameLift Streams のタイムアウト値。
- 接続タイムアウト: クライアントがストリームセッションに接続または再接続するためのタイムアウト値。
- アイドルタイムアウト: ユーザー入力なしでストリームセッションをアイドル状態にできる最大時間。
- セッション長タイムアウト: ストリームセッションの最大時間。

各タイムアウトとその可能な値の詳細については、「」を参照してください[ストリームセッションに影響するタイムアウト値](#)。

Proton for Amazon GameLift Streams との互換性のテストとトラブルシューティング

Amazon GameLift Streams アプリケーションが Proton ランタイム環境で実行されている場合、このセクションはアプリケーションと Proton レイヤー間の互換性の問題をトラブルシューティングするのに役立ちます。これらの手順には、Amazon GameLift Streams が使用する環境をシミュレートして、Proton を独自のマシンにインストールする一連のスクリプトが含まれています。Amazon GameLift Streams サービスを使用せずにトラブルシューティングを行うことで、アプリケーションとランタイム環境に固有の問題のトラブルシューティングに集中できます。

テストとトラブルシューティングの大まかな手順

1. Ubuntu 22.04 LTS マシンを取得します。ローカルマシンまたは Amazon EC2 クラウドベースのデスクトップを使用できます。手順については、以下のトピックから選択してください。
 - [ローカルマシンをセットアップする](#)
 - [リモートマシンをセットアップする](#)
2. Proton ランタイム環境をインストールして、アプリケーションをテストおよびデバッグします。ガイダンス[Proton でのトラブルシューティング](#)については、「」を参照してください。

Proton の既知の問題

最新の互換性とトラブルシューティングリソースについては、[Proton GitHub Wiki](#) を参照してください。Proton GitHub 問題 [トラッカーで問題](#)を検索することもできます。以下は、お客様が Proton で Windows アプリケーションを実行するときに遭遇したことに注意すべき特定の問題です。

Proton での Godot アプリケーション

- Proton で実行されている Godot ベースのアプリケーションでは、Amazon Vulkan キャプチャレイヤーが有効になっている場合、黒い画面が表示されることがあります。この問題を軽減するには、環境変数 `VK_LAYER_AMZN_BLITSURFACE_SHARED_TEXTURES=0` を設定して、ストリーミング時に共有テクスチャを無効にします。

Proton での Unreal Engine アプリケーション

- Proton 8.x with Electra Media Player (Unreal Engine プラグイン) で問題が発生した場合は、<https://github.com/ValveSoftware/wine/pull/257> にある修正を使用することをお勧めします。

Proton をトラブルシューティングするためのローカルマシンのセットアップ

Proton は、Windows アプリケーションを Linux で実行できるようにする互換性レイヤーです。そのため、テストとトラブルシューティングを行う Ubuntu マシンが必要です。ローカル Ubuntu マシンがない場合は、Amazon EC2 を使用してリモートマシンを設定できます。これを行うには、[リモートマシンをセットアップする](#)代わりに「」のステップに従います。

前提条件

- [Ubuntu 22.04 LTS](#)。インストール手順については、Ubuntu の [Install Ubuntu Desktop](#) チュートリアルを使用できます。
- NVIDIA GPU

GPU ドライバーをインストールする

最新の GPU ドライバーをインストールすると、アプリケーションのパフォーマンスの低下やクラッシュを防ぐことができます。

システムが使用する GPU ドライバーを確認するには

1. ターミナルで次のコマンドを実行します。

```
lshw -C display | grep driver
```

2. 正しいドライバーがインストールされている場合は、次の出力が表示されます。<*gpu*> は NVIDIA nvidia用です。 configuration: driver=<gpu> latency=0

最新の NVIDIA GPU ドライバーをインストールするには

[NVIDIA ドライバーのインストール](#)の手順に従います。

GPU ドライバーを検証する

GPU ドライバーがインストールされ、正しく動作していることを確認します。これを検証する方法の 1 つは、ターミナルで [vkcube](#) アプリケーションを実行することです。

1. 次のコマンドを使用してvulkan-tools、apt パッケージをインストールします。

```
sudo apt install -y vulkan-tools
```

2. vkcube を実行します。

3. 出力を確認します。

- システムが正しく正しい GPU を使用している場合は、GPU の名前とともに次のような出力が表示されます。 Selected GPU 0: AMD Radeon Pro V520 (RADV NAVI12), type: 2
- アプリケーションが GPU を正しく使用できない場合は、次のような異なる出力が表示されることがあります。 Selected GPU 0: llvmpipe (LLVM 15.0.7, 256 bits), type: 4

この場合、GPU ドライバーを確認し、必要に応じて再インストールします。

次のステップ

ローカル Ubuntu マシンの準備ができたら、次のステップは Proton をセットアップすることです。手順については、「」を参照してください[Proton でのトラブルシューティング](#)。

リモート Amazon EC2 マシンをセットアップして Proton をトラブルシューティングする

ローカル Ubuntu マシンがない場合は、以下の手順に従ってリモートマシンをセットアップします。

このステップでは、Amazon Elastic Compute Cloud (Amazon EC2) を使用してリモート Ubuntu マシンを設定します。これを使用して、アプリケーションの Proton for Amazon GameLift Streams との互換性をトラブルシューティングします。このトピックでは、Ubuntu 22.04 LTS、必要な GPU ドライバー、ビジュアルリモートデスクトップ用の Amazon DCV サーバーを使用して Amazon EC2 インスタンスを設定する方法について説明します。

Ubuntu 22.04 LTS AMI で Amazon EC2 インスタンスを起動する

1. AWS マネジメントコンソールで Amazon EC2 に移動します。
2. インスタンスの起動を選択します。
3. 名前に「Amazon GameLift ストリームテスト」と入力します。
4. アプリケーションイメージと OS イメージ (Amazon マシンイメージ) に Ubuntu Server 22.04 LTS (HVM) を選択します。
5. インスタンスタイプの g4dn.2xlarge を選択します。
6. キーペア (ログイン) で、SSH を使用してインスタンスにアクセスする場合はキーペアを選択します。を使用してインスタンスに接続するには、AmazonSSMManagedInstanceCoreポリシーでインスタンスプロファイルを使用することをお勧めします AWS Systems Manager Session Manager。詳細については、[「既存の IAM ロールへの Session Manager アクセス許可の追加」](#) を参照してください。
7. ネットワーク設定で、新しいセキュリティグループを作成します。
8. セキュリティグループ名に DCV と入力します。
9. Amazon DCV を使用したアクセスを許可する Anywhere には Custom TCP、タイプ、ポート範囲 8443、およびソースタイプのインバウンドセキュリティグループルールを追加します。
- 10.ストレージを少なくとも 256GB に増やし、ストレージタイプとして gp3 を選択します。
- 11.[インスタンスの作成] を選択してください。

これでインスタンスが起動します。

Linux [インスタンスに接続する](#) の手順に従って、SSH または を使用してインスタンスに接続します AWS Systems Manager Session Manager。

GPU ドライバーをインストールする

G4dn - NVIDIA GPU

次のコマンドを実行して、追加のモジュールと Linux ファームウェアをインストールします。

```
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
```

「Linux に NVIDIA ドライバーをインストールする」の「Ubuntu および Debian 用の NVIDIA GRID ドライバー」の手順に従います。

ユーザー環境をセットアップする

次のコマンドを実行して GPU を使用できるようにユーザー環境を設定します。これにより、次のことが実行されます。

- ビデオデバイスへのアクセスを許可するvideoグループと、レンダリングデバイスへのアクセスを許可するrenderグループを追加します。
- NVIDIA ドライバーと AWS CLI Amazon S3 からアプリケーションまたはゲームをダウンロードするため必要なをインストールします。

```
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
```

Amazon DCV のインストールと設定

SSH または を使用してインスタンスに再接続 AWS Systems Manager Session Manager し、Linux for Ubuntu に Amazon DCV サーバーをインストールするの指示に従います。

- ドキュメントの説明に従ってサーバーが正しく設定されていることを確認します。
- [「NVIDIA GPU 用の NVIDIA ドライバーのインストールと設定」](#) のステップに従います。
- 「[サーバーのインストール](#)」ガイドのステップ 7 (Ubuntu タブに移動) で説明されているように、Amazon DCV ユーザーをビデオグループに追加します。

Amazon DCV サーバーのオプションパーツをインストールする必要はありません。

完了したら、次のコマンドを実行して Amazon DCV サーバーを起動します。

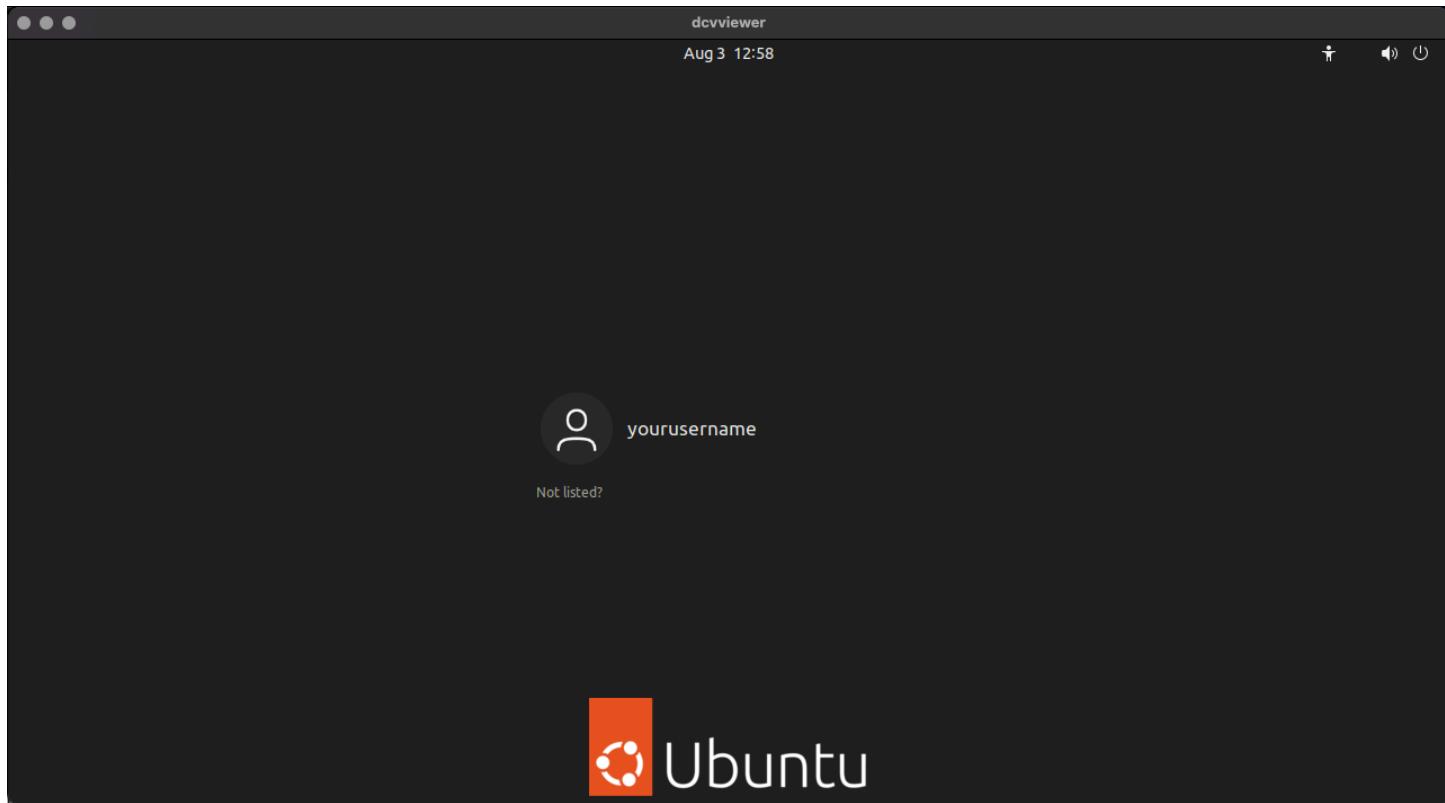
```
sudo systemctl start dcvserver  
sudo systemctl enable dcvserver
```

Amazon DCV クライアントを使用した Ubuntu Server への接続

Ubuntu インスタンスに再接続し、以下を実行してユーザーのセッションを作成します。

```
sudo dcv create-session --owner user --user user my-session --type console
```

Amazon DCV クライアントを使用して、パブリック IP アドレスを使用して Ubuntu インスタンスにアクセスできるようになりました。Amazon DCV クライアントを起動すると、ウィンドウが表示され、ビジュアルディスプレイから Ubuntu インスタンスにアクセスできます。



GPU ドライバーを検証する

GPU ドライバーがインストールされ、正しく動作していることを確認します。これを検証する 1 つの方法は、ターミナルで [vkcube](#) アプリケーションを実行することです。

1. 次のコマンドを使用して vulkan-tools、apt パッケージをインストールします。

```
sudo apt install -y vulkan-tools
```

2. vkcube を実行します。

3. 出力を確認します。

- システムが正しく正しい GPU を使用している場合は、GPU の名前とともに次のような出力が表示されます。Selected GPU 0: AMD Radeon Pro V520 (RADV NAVI12), type: 2
- アプリケーションが GPU を正しく使用できない場合は、次のような異なる出力が表示されることがあります。Selected GPU 0: llvmpipe (LLVM 15.0.7, 256 bits), type: 4

この場合、GPU ドライバーを確認し、必要に応じて再インストールします。

Podman のセットアップ (Proton のみ)

Proton ランタイムを使用している場合は、Proton のビルドプロセスで使用されるコンテナである [Podman](#) をインストールする必要があります。ターミナルで次の手順を実行します。

1. Proton のビルドプロセスが使用するコンテナである Podman をインストールします。

```
sudo apt install podman
```

2. ファイル/etc/subgidと /etc/subgid

- a. ファイルで Linux マシンのユーザー名とパスワードが一覧表示されていることを確認します。ファイルを開くか、 cat コマンドを使用してファイルの内容を確認できます。形式例: test:165536:65536。はユーザー名testに対応します。
- b. リストにない場合は、追加します。形式例: test:165536:65536。はユーザー名testに対応します。

```
$ cat /etc/subuid
ceadmin:100000:65536
test:165536:65536

$ cat /etc/subgid
ceadmin:100000:65536
test:165536:65536
```

詳細については、[Podman のドキュメントの「基本的なセットアップとルートレス環境での Podman の使用」](#)を参照してください。

次のステップ

Amazon GameLift Streams との互換性の問題をトラブルシューティングするための Amazon EC2 インスタンスと環境のセットアップが完了しました。次のステップでは、Proton をセットアップします。手順については、「」を参照してください[Proton でのトラブルシューティング](#)。

Proton での互換性のトラブルシューティング

このステップでは、独自のマシンに Proton をセットアップして、Amazon GameLift Streams アプリケーションと Proton 間の互換性の問題をトラブルシューティングできるようにします。Amazon

GameLift Streams サーバーを使用せずにシミュレートされた環境でアプリケーションを実行すると、アプリケーションとランタイム環境に固有の問題を特定するのに役立ちます。

前提条件

- GPU ドライバーがインストールされた Ubuntu 22.04 LTS。手順については、[ローカルマシンをセットアップする](#)「」または「」を参照してください[リモートマシンをセットアップする](#)。

Proton をインストールする

Ubuntu 22.04 LTS マシンに Proton をインストールするには、次のスクリプトを使用して、Proton [GitHub リポジトリからテストする Proton](#) のバージョンをクローン、ビルド、設定します。

1. 次のコードをコピーして、Ubuntu 22.04 LTS マシン `proton-setup.sh` のというファイルに貼り付けます。

```
#!/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. このステップでは、Proton セットアップスクリプトを実行して Proton と追加の依存関係をクローンしてインストールします。スクリプトは、インストールする Proton バージョンのタグまたはブランチ名を引数として受け入れます。Amazon GameLift Streams が提供する Proton のカスタムビルドのいずれかをシミュレートするには、以下のそのバージョンの手順を使用します。

 Note

GitHub からのクローン作成には時間がかかることが予想されます。ダウンロードするサブモジュールが多数あり、合計は数ギガバイトです。

ターミナルで、`proton-setup.sh`スクリプトを実行し、Proton バージョンブランチを指定します。

- 組み込み Proton バージョン

- Proton 9.0-2 (PROTON-20250516) の場合は、[experimental-9.0-20241121b](#) を使用します。

```
proton-setup.sh experimental-9.0-20241121b
```

- Proton 8.0-5 (PROTON-20241007) の場合は、[experimental-8.0-20240205](#) を使用します。

```
proton-setup.sh experimental-8.0-20240205
```

通常、追加のソースコードは必要ありません。ただし、Electra Media Player (Unreal Engine プラグイン) で問題が発生した場合は、<https://github.com/ValveSoftware/wine/pull/257> にある修正を使用することをお勧めします。

 Note

Proton 8.0-2c (PROTON-20230704) の場合、Amazon GameLift Streams は独自のビルドを使用しますが、ローカルでボールドすることはできません。

- 推奨されるカスタム Proton バージョン

カスタム Proton バージョンの場合は、Proton experimental_8.0 ブランチを使用することをお勧めします。

```
proton-setup.sh experimental_8.0
```

- その他のカスタム Proton バージョン

他の Proton バージョンでは、[Proton リリース](#)に記載されている正確なブランチ名またはタグ名を使用します。

```
proton-setup.sh branch-or-tag-name
```

インストールが成功した場合、ターミナルの出力は次のようにになります。

```
...
```

```
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
```

次のステップで Proton を実行する必要があるため、出力から次の変数を書き留めます。

- Proton バイナリの場所
- STEAM_COMPAT_DATA_PATH
- STEAM_COMPAT_CLIENT_INSTALL_PATH

Proton でアプリケーションを実行する

次の手順では、アプリケーション実行可能ファイルが にあることを前提としています path/
myapplication/bin/application.exe。アプリケーションのパスとファイル名に置き換えま
す。

- ターミナルで、アプリケーション実行可能ファイルがあるフォルダに移動します。

```
cd path/myapplication/bin/application.exe
```

- Proton でアプリケーションを実行します。Proton バイナリの場所と、前のステップで取得した環
境変数を使用します。

```
STEAM_COMPAT_DATA_PATH=/home/test/protonBuild/dist/compatdata
STEAM_COMPAT_CLIENT_INSTALL_PATH=anything /home/test/protonBuild/dist/proton run
application.exe
```

これで、アプリケーションは起動を試みるはずです。アプリケーションがローカルで起動
し、Amazon GameLift Streams では起動しない場合、Amazon GameLift Streams APIs を呼び出す際
の設定の問題が原因である可能性があります。API コールパラメータが正しいことを確認します。そ
れ以外の場合は、デバッグの次のステップに進みます。

ログファイルを使用してアプリケーションをデバッグする

ローカル Proton 環境で実行されているアプリケーションに問題がある場合は、出力ログを確認しま
す。ログには、アプリケーションとランタイム環境からの出力が含まれます。アプリケーション側で
アプリケーションが問題を検出できないトレース。

ログ出力をなどのテキストファイルにダンプするにはproton.log、次のコマンドを使用します。

```
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
```

また、Proton は、問題の原因が、" プラグイン、実装されていない関数、dlls の欠落などであるかどうかも示します。詳細については、[「Quine HQ の Debugging」guide](#) を参照してください。アプリケーション側で修正できない Proton または" エラーがログに見つかった場合は、AWS アカウントマネージャーに問い合わせるか、[AWS re:Post](#) に質問を投稿して、さらなるデバッグに役立ててください。

Unreal Engine のパフォーマンスのプロファイリング

このセクションでは、Unreal Engine ゲームまたはアプリケーションのパフォーマンスを分析する方法について説明します。これにより、最適化するエリアのを特定し、Amazon GameLift Streams でのストリーミングをスムーズにすることができます。

Unreal Engine のコンソールとその組み込みの統計コマンドを使用して、ゲームのパフォーマンスの詳細を確認できます。コンソールには、出荷不可能なビルドまたはエディタでアクセスできます。出荷不可能なビルドとは、デバッグまたは開発設定を使用して構築されたプロジェクトを指します。

コンソールにアクセスするには

出荷不可能なビルドおよび[エディタで再生](#)モードでは、チルダ (~) キーを押してコンソールを開きます。チルドキーをダブルクリックしてコンソールを展開します。

コンソールを使用するためのヒントをいくつか紹介します。

- キーワードを入力して、そのキーワードを含むすべての可能なコマンドを一覧表示します。矢印キーを使用してリストをスクロールします。
- 矢印キーまたはページアップキーとページダウンキーを使用して履歴をスクロールします。
- ログはプロジェクトの Saved/Logs ディレクトリの.txt ファイルに保存されます。

ゲームのパフォーマンスをプロファイリングするには

- まず、コマンドstat fpsとstat unit コマンドを実行します。これにより、ゲームがパフォーマンスに苦労する場所の概要がわかります。

- `stat fps`: 1 秒あたりの現在のフレームを表示します。
- `stat unit`: フレームを複数のサブセクションに分割します。
 - フレーム: フレームのシミュレーションが開始されてからフレームの表示が画面に表示されるまでの合計ウォールクロック時間。
 - ゲーム: フレームあたりのゲームシミュレーションスレッドにかかる合計 CPU 時間。
 - 描画: レンダリングスレッドがシーンを GPU のコマンドに変換して GPU に送信する合計 CPU 時間。
 - GPU: GPU がすべてのコマンドを処理する合計時間。
 - Draws: フレームに送信された描画の合計数。
 - プリム: 描画された三角形の総数。
- 2. ゲームをプレイして、パフォーマンスが低い領域を特定します。これは、ゲーム、描画、または GPU で FPS が減り、時間が長くなったことを示します。
- 3. を実行して`stat game`、さまざまなゲームプレイグループの所要時間を確認します。
- 4. AI、アニメーション、物理、ゲームプレイ、スクリプティングなどの特定のゲームプレイ要因の統計を絞り込みます。ここにいくつか例を挙げます。
 - `stat ai`: AI の動作を計算する時間。
 - `stat anim`: スキンメッシュを計算する時間。
 - `stat physics`: 物理シミュレーションを計算する時間。
- 5. を実行して`stat drawcount`、どのレンダリング領域が最も描画を生成するかを確認します。このリストには、が出力するレンダーパスと、各フレームに出力されるドローの数が表示されます。次のステップで GPU 統計を分析することで、詳細情報を取得できます。
- 6. を実行して`stat gpu`、GPU 時間を最も多く消費するレンダータイプを確認します。
- 7. レンダリングタイプを照明、シャドウ、内包部(照明)、髪、後処理など、幅広いグループに絞り込みます。一般的な例をいくつか示します。
 - `stat lightrendering`: ライトとシャドウをレンダリングする GPU 時間。
 - `stat shadowrendering`: さまざまなシャドウを更新する GPU 時間。
 - `stat scenerendering`: シーンをレンダリングする GPU 時間。

このセクションでは、使用可能なコマンドのサブセットのみについて説明します。ゲームの機能に応じて、アセットストリーミング、仮想テクスチャリング、CPU タスクワークロード分散、スレッ

ディング、サウンド、パーティクルなどの分野の統計を調べます。詳細については、[「Stat コマンド」](#)を参照してください。

リージョン、クオータ、制限

Amazon GameLift Streams は複数の AWS リージョン、IPv4 と IPv6 の両方の接続をサポートするデュアルスタックサービスエンドポイントを提供します。このサービスは、米国東部(オハイオ)、米国西部(オレゴン)、アジアパシフィック(東京)、欧州(フランクフルト)などの主要なロケーションから動作し、レイテンシーとストリーム品質を最適化するために、合計でリモートロケーションと呼ばれる追加の AWS リージョンおよびロケーションを管理できます。

サービスインフラストラクチャは、3 つの主な制約カテゴリによって管理されます。

- サービスクォータ
- API レート制限
- サービスの制限を修正

これには、アプリケーションサイズ、リージョンあたりのアプリケーション数、ファイル管理容量、さまざまなストリームクラスとリージョンにわたる GPU 割り当ての制限が含まれます。このサービスは、1 秒あたり 1~20 リクエストのさまざまなオペレーションに特定の API レート制限を実装し、安定したサービスパフォーマンスを確保します。さらに、ストリームグループ設定、GPU デプロイ、アプリケーションの関連付けには、すべてのお客様に均一に適用される固定サービスの制限があります。

AWS リージョン Amazon GameLift Streams でサポートされている およびリモートロケーション

AWS リージョンは、地理的領域の AWS リソースのコレクションです。各 AWS リージョンは他のリージョンから分離され、独立しています。の一般的な情報については AWS リージョン、『』の「[の管理 AWS リージョン](#)」を参照してください AWS 全般のリファレンス。

次の表に、Amazon GameLift Streams サービスが利用可能な AWS リージョンと、各リージョンのエンドポイントを示します。Amazon GameLift Streams コンソールで作業するか、(AWS CLI) を使用する AWS Command Line Interface か、プログラムによる呼び出しを行うかにかかわらず、指定されたリージョンにすべての Amazon GameLift Streams アプリケーションおよびストリームグループリソースを作成します。これらのリソースを作成するリージョンは、プライマリロケーションと呼ばれます。プライマリロケーションのエンドポイントを使用して、プログラムで Amazon GameLift Streams サービスに接続します。

サービスエンドポイント

Amazon GameLift Streams では、デュアルスタックのサービスエンドポイントがサポートされます。そのため、クライアントとリソースは IPv6 または IPv4 を使用してサービスとやり取りできます。

リージョン名	リージョン	エンドポイント	プロトコル
米国東部 (オハイオ)	us-east-2	gameliftstreams.us-east-2.api.aws	HTTPS
米国西部 (オレゴン)	us-west-2	gameliftstreams.us-west-2.api.aws	HTTPS
アジアパシフィック (東京)	ap-northeast-1	gameliftstreams.ap-northeast-1.api.aws	HTTPS
欧州 (フランクフルト)	eu-central-1	gameliftstreams.eu-central-1.api.aws	HTTPS

リモート口케ーション

Amazon GameLift Streams は、リモート口ケーションにカバレッジを拡張できるため、より多くの口ケーションでアプリケーションとストリームセッションをホストできます。使用できるリモート口ケーションは、プライマリ口ケーションによって異なります。レイテンシーとストリーム品質を最適化するために、地理的にユーザーに近い場所を選択することをお勧めします。

プライマリロケーション	リモートロケーション		
米国東部 (オハイオ) – us-east-2	<ul style="list-style-type: none">米国東部 (バージニア北部) – us-east-1米国西部 (オレゴン) – us-west-2欧州 (アイルランド) – eu-west-1欧州 (フランクフルト) – eu-central-1アジアパシフィック (東京) – ap-northeast-1		
米国西部 (オレゴン) – us-west-2	<ul style="list-style-type: none">米国東部 (バージニア北部) – us-east-1米国東部 (オハイオ) – us-east-2欧州 (アイルランド) – eu-west-1欧州 (フランクフルト) – eu-central-1アジアパシフィック (東京) – ap-northeast-1		
アジアパシフィック (東京) – ap-northeast-1	<ul style="list-style-type: none">米国東部 (バージニア北部) – us-east-1		

プライマリロケーション	リモートロケーション
	<ul style="list-style-type: none">米国西部 (オレゴン) – us-west-2米国東部 (オハイオ) – us-east-2欧州 (アイルランド) – eu-west-1欧州 (フランクフルト) – eu-central-1米国東部 (バージニア北部) – us-east-1米国西部 (オレゴン) – us-west-2米国東部 (オハイオ) – us-east-2欧州 (アイルランド) – eu-west-1アジアパシフィック (東京) – ap-northeast-1
欧州 (フランクフルト) – eu-central-1	

Amazon GameLift Streams のストリームクラスでサポートされているロケーション

次の表は、サポートされているすべてのロケーション AWS リージョンとリモートロケーションにおける各ストリームクラスファミリーの可用性を示しています。

リージョン名	リージョン	gen6*	gen5*	gen4*
アジアパシフィック (東京)	ap-northeast-1	✓ は い	✓ は い	✓ は い
欧州 (フランクフルト)	eu-central-1	✓ は い	✓ は い	✓ は い
欧州 (アイルランド)	eu-west-1	✗ い いえ	✓ は い	✓ は い
米国東部 (バージニア北部)	us-east-1	✓ は い	✓ は い	✓ は い
米国東部 (オハイオ)	us-east-2	✓ は い	✓ は い	✓ は い
米国西部 (オレゴン)	us-west-2	✓ は い	✓ は い	✓ は い

Amazon GameLift Streams サービスクォータ

Service Quotas (制限とも呼ばれます) は、 AWS アカウントのサービスリソースまたはオペレーションの最大数です

Amazon GameLift Streams のサービスクォータの多くは、アカウントでストリーミングするために設定できる GPUs (コンピューティングリソース) の総数を制限します。具体的には、これらの GPU サービスクォータは、アカウント内のすべてのストリームグループでロケーションごとにリクエストできる特定のストリームクラスファミリーの GPUs の最大数を指定します。たとえば、アカウントに 5 つの gen5n GPUs の制限がある場合 us-west-2、すべてのストリームグループ us-west-2 の合計ストリーム容量を提供するために必要な gen5n GPUs の合計は 5 以下である必要があります。これには、常時オン容量とオンデマンド容量の両方の GPUs が含まれます。

クォータがストリーム容量とやり取りする方法の詳細については、「」を参照してください [キャパシティとサービスクォータ](#)。また、Amazon GameLift Streams で注意すべき追加の制限 [その他の制限事項](#)については、必ず [API レート制限](#) を確認してください。

AWS サービスとして GameLift Streams を選択して、Service Quotas コンソールでデフォルトまたは適用されたアカウントレベルのクオータと使用率を表示します。

サービスクオータの一般的な情報については、の[AWS 「サービスクオータ」](#)を参照してくださいAWS 全般のリファレンス。

サービスクオータ

次の表では、GPU クオータはデフォルトですべて 0 です。ただし、お使いのアカウントに適用されているクオータは異なっている場合があります。確認するには、にサインイン AWS マネジメントコンソールし、[Amazon GameLift Streams](#) に Service Quotas コンソールを開きます。ここでは、適用されたアカウントレベルのクオータ値列で現在のクオータを確認し、使用率列でこれらのクオータの使用率を確認し、これらの値を増やすリクエストを送信します。

名前	デフォルト	引き上げ可能	説明
アプリケーションサイズ (GiB)	サポートされている各リージョン: 100	可能	このアカウントでのアプリケーションの最大合計サイズ (GiB)。ギガバイト (GiB) は、 $1024 \times 1024 \times 1024$ バイトに等しいことに注意してください。
アプリケーション	サポートされている各リージョン: 20	可能	AWS リージョンごとに、このアカウントで作成できるアプリケーションの最大数。
アプリケーションあたりのファイル数	サポートされている各リージョン: 30,000	あり	このアカウントで 1 つのアプリケーションに保持できるファイルの最大数。

名前	デフォルト	引き上げ可能	説明
Gen4n GPU、ap-northeast-1	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループで ap-northeast-1 リージョンでストリーミング用に設定できる Gen4n GPUs の最大数。「Gen4n_high」などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。
Gen4n GPU、eu-central-1	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループで eu-central-1 リージョンでストリーミング用に設定できる Gen4n GPUs の最大数。「Gen4n_high」などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。

名前	デフォルト	引き上げ可能	説明
Gen4n GPU、eu-west-1	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループにわたって eu-west-1 ロケーションでストリーミングするために設定できる Gen4n GPUs の最大数。「Gen4n_high」などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。
Gen4n GPU、us-east-1	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループにわたって us-east-1 ロケーションでストリーミングするために設定できる Gen4n GPUs の最大数。「Gen4n_high」などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。

名前	デフォルト	引き上げ可能	説明
Gen4n GPU、us-east-2	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループにわたって us-east-2 ロケーションでストリーミングするために設定できる Gen4n GPUs の最大数。「Gen4n_high」などのマルチテナントストリームクラスは、GPUごとに複数のセッションのストリーミングをサポートします。
Gen4n GPU、us-west-2	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループにわたって us-west-2 ロケーションでストリーミングするために設定できる Gen4n GPUs の最大数。「Gen4n_high」などのマルチテナントストリームクラスは、GPUごとに複数のセッションのストリーミングをサポートします。

名前	デフォルト	引き上げ可能	説明
Gen5n GPU、ap-northeast-1	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループの ap-northeast-1 ロケーションでストリーミング用に設定できる Gen5n GPUs の最大数。"Gen5n_high" などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。
Gen5n GPU、eu-central-1	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループで eu-central-1 ロケーションでストリーミング用に設定できる Gen5n GPUs の最大数。"Gen5n_high" などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。

名前	デフォルト	引き上げ可能	説明
Gen5n GPU、eu-west-1	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループにわたって eu-west-1 ロケーションでストリーミングするために設定できる Gen5n GPUs の最大数。"Gen5n_high" などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。
Gen5n GPU、us-east-1	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループにわたって us-east-1 ロケーションでストリーミングするために設定できる Gen5n GPUs の最大数。"Gen5n_high" などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。

名前	デフォルト	引き上げ可能	説明
Gen5n GPU、us-east-2	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループにわたって us-east-2 ロケーションでストリーミングするために設定できる Gen5n GPUs の最大数。"Gen5n_high" などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。
Gen5n GPU、us-west-2	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループにわたる us-west-2 ロケーションでのストリーミング用に設定できる Gen5n GPUs の最大数。"Gen5n_high" などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。

名前	デフォルト	引き上げ可能	説明
Gen6n GPUs、ap-northeast-1	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループで ap-northeast-1 ロケーションでストリーミング用に設定できる Gen6n GPUs の最大数。"Gen6n_high" などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。
Gen6n GPUs、eu-central-1	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループで eu-central-1 ロケーションでストリーミング用に設定できる Gen6n GPUs の最大数。"Gen6n_high" などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。

名前	デフォルト	引き上げ可能	説明
Gen6n GPUs、us-east-1	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループにわたって us-east-1 ロケーションでストリーミングするために設定できる Gen6n GPUs の最大数。"Gen6n_high" などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。
Gen6n GPUs、us-east-2	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループにわたって us-east-2 ロケーションでストリーミングするために設定できる Gen6n GPUs の最大数。"Gen6n_high" などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。

名前	デフォルト	引き上げ可能	説明
Gen6n GPUs、us-west-2	サポートされている各リージョン: 0	<u>あり</u>	このアカウントのすべてのストリームグループにわたる us-west-2 ロケーションでのストリーミング用に設定できる Gen6n GPUs の最大数。"Gen 6n_high" などのマルチテナントストリームクラスは、GPU ごとに複数のセッションのストリーミングをサポートします。
ストリームグループ	サポートされている各リージョン: 5	<u>あり</u>	このアカウントでリージョンごとに作成できるストリームグループの最大数 AWS。ストリームグループは、アプリケーションをエンドユーザーにストリーミングするコンピューティングリソースのコレクションです。

Amazon GameLift Streams API レート制限

これらの制限は、から の Amazon GameLift Streams サービス AWS アカウントへの 1 秒あたりのリクエストの最大レートを反映しています AWS リージョン。

API オペレーション:	1 秒あたりのリクエスト
AddStreamGroupLocations	5
AssociateApplications	5
CreateApplication	5
CreateStreamGroup	1
CreateStreamSessionConnection	20
DeleteApplication	5
DeleteStreamGroup	5
DisassociateApplications	5
ExportStreamSessionFiles	20
GetApplication	10
GetStreamGroup	10
GetStreamSession	20
ListApplications	10
ListStreamGroups	10
ListStreamSessions	20
ListStreamSessionsByAccount	20
ListTagsForResource	10
RemoveStreamGroupLocations	5
StartStreamSession	20
TagResource	10

API オペレーション:	1 秒あたりのリクエスト
TerminateStreamSession	20
UntagResource	10
UpdateApplication	5
UpdateStreamGroup	5

その他の Amazon GameLift Streams の制限事項

このページには、ストリーミングソリューションを作成する際に注意すべき他の制限事項が記載されています。これらの制限は、すべてのお客様の サービス内で修正されます。

名前	制限	説明
ストリームグループ内のアプリケーション	100	ストリームグループに関連付けることができる Amazon GameLift Streams アプリケーションの最大数。
ストリームグループの GPUs	2500	すべてのリージョンとリモートロケーションにおけるストリームグループ内の GPUs の最大数。
単一ファイルサイズ (GiB)	80 GiB	アプリケーション内の 1 つのファイルの最大サイズ (GiB)。ギガバイト (GiB) は、1024 x 1024 x 1024 バイトに等しいことに注意してください。
アプリケーションあたりのストリームグループの関連付け	100	Amazon GameLift Streams アプリケーションが関連付け

名前	制限	説明
		けることができるストリームグループの最大数。

Amazon GameLift Streams の使用状況と請求の管理

このトピックでは、Amazon GameLift Streams の使用状況、コスト、請求をモニタリングおよび管理して、ストリーミング費用を最適化する方法について説明します。

以下の情報については、Amazon GameLift Streams の[料金ページ](#)も参照してください。

- コスト内訳: Amazon GameLift Streams を使用する際の AWS 料金について説明します。
- Amazon GameLift Streams 料金: Amazon GameLift Streams のコストを確認し、さまざまなオプションを比較します。
- ストリームキャパシティ予約: 事前に計画し、顧客の需要を満たすのに十分なストリームキャパシティがあることを確認します。

Amazon GameLift Streams の請求書と使用状況を確認する

AWS コンソールまたは AWS Billing and Cost Management ツールを使用して、Amazon GameLift Streams の請求と使用状況を確認できます AWS CLI。

AWS コンソールから請求書を表示するには、AWS Billing 「[ユーザーガイド](#)」の「[請求書の表示](#)」を参照してください。

を通じて請求を表示するには AWS CLI、Billing and Cost Management API [GetCostAndUsage](#) を使用して呼び出します。たとえば、次のコマンドを使用して Amazon GameLift Streams の毎月の請求書を取得し、日付を関連するものに置き換えます。

Example : **GetCostAndUsage** API を使用して請求書を表示する

```
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
```

ここで、などのフィルターはAmazon GameLift Streams-bill-filter.json、Amazon GameLift Streams サービスを次のように指定します。

```
{  
    "Dimensions": {  
        "Key": "SERVICE",  
        "Values": ["Amazon Amazon GameLift Streams"]  
    }  
}
```

Amazon GameLift Streams のコストを管理するためのベストプラクティス

予期しないコストを避けるため、Amazon GameLift Streams のコストを管理するには、次のツールと手法を使用することを強くお勧めします。

請求アラートを作成して、使用状況をモニタリングする

AWS Budgets を使用して請求アラートを設定します。これにより、コストと使用状況を追跡し、アラートに迅速に対応して予期しないコストを回避できます。予算内に収まるようにアクションをトリガーするように請求アラートを設定することもできます。デフォルトでは、予算にはすべての AWS サービスが含まれます。Amazon GameLift Streams のみの予算を指定するには、[予算フィルター](#)を追加します。

詳細については、以下の各トピックを参照してください。

- [予算の作成](#)
- [AWS Budgets のベストプラクティス](#)

ストリームグループを容量ゼロにスケールする

割り当てられたストリーム容量には、現在ストリームセッションをホストしていない場合でも、引き続きコストが発生します。不要なコストを避けるため、ストリームグループを使用しない場合は容量をゼロにスケールします。これにより、ストリームグループでリソースが割り当てられなくなります。常時オンストリームとオンデマンドストリームの容量をゼロに設定すると、接続されているすべてのストリームが終了します。準備ができたら、容量をスケールアップしてストリームグループを再利用できます。

手順については、「」を参照してください[容量の編集](#)。

⚠ Warning

ストリームグループを再度使用する予定がない限り、ストリームグループを削除しないでください。ストリームグループを削除すると、元のストリームグループを復元できず、新しいストリームグループを作成する必要があります。

元のアプリケーションファイルを削除する

ストレージコストを最適化するために、Amazon S3 バケットにアップロードした元のアプリケーションファイルを削除できます。アプリケーションのステータスが準備完了の場合は、ファイルを削除しても安全です。この時点で、Amazon GameLift Streams にはアプリケーションファイルのスナップショットがあり、元のファイルにアクセスできなくなります。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。